
6,– DM Ös 50,– sfr 6,–
Das führende Schneider-Magazin
CPC
Schneider
INTERNATIONAL
8
August 1986
2. Jahrgang
Exclusiv:
GRAFIK MIT JOYCE
Aktuell:
AMSTRAD-SHOW ZEIGT
NEUE WEGE
SCHNEIDERWARE
RS-232-Schnittstelle
Pascal:
LISTER II
Tips & Tricks:
MUSIKINTERRUPT
VARDUMP
Programme:
PAINT MASTER
ORTWINS RUN
Im Test:
SELBSTBAU-PLOTTER
ECB-BUS
DER CLOU:
FREMDMONITORE
AM CPC

Impressum

Schneider CPC International
erscheint in der
DMV - Daten & Medien Verlagsges. mbH

Chefredakteur
Christian Widuch (verantwortlich)

Stellvertretender Chefredakteur
Stefan Ritter

Redaktion
Thomas Morgen (TM), Michael Ebbrecht (ME), Heinrich Stiller (HS), Michael Ceol (MC)

Satz
Silvia Führer, Renate Wells

Gestaltung
Renate Wells, Gerd Köberich

Grafik/Illustration
Heinrich Stiller

Fotografie
Gerd Köberich

Anzeigenleitung
Wolfgang Schnell
Es gilt die Anzeigenpreisliste Nr. 2 vom 1.1.1986

Freie Mitarbeiter
Horst Franke (HF), P. Richter, J. Hüpper, K.R. Meißner, J. Keller, M. Anton, V. Reher, Dipl.-Ing. H. Scheruhn, V. Klug, T. Schlote, M.E. Kahle, T. Fippl, M. Uphoff, J. Schwarze, T. Kochmann, R. Nitsche, S. Strathaus, E. Röscheisen

Anschrift Verlag/Redaktion
DMV - Daten & Medien Verlagsges. mbH
Postfach 250, Fuldaer Str. 6
3440 Eschwege
Telefon: 05651/8702
Telex 993 210 dmv d

Vertrieb
Verlagsunion
Friedrich-Bergius-Straße 20
6200 Wiesbaden

Vertrieb Österreich
Pressegroßvertrieb
Salzburg Ges.mbH & Co. KG
Niederalm 300
5081 Anif
Tel.: 06246/3721

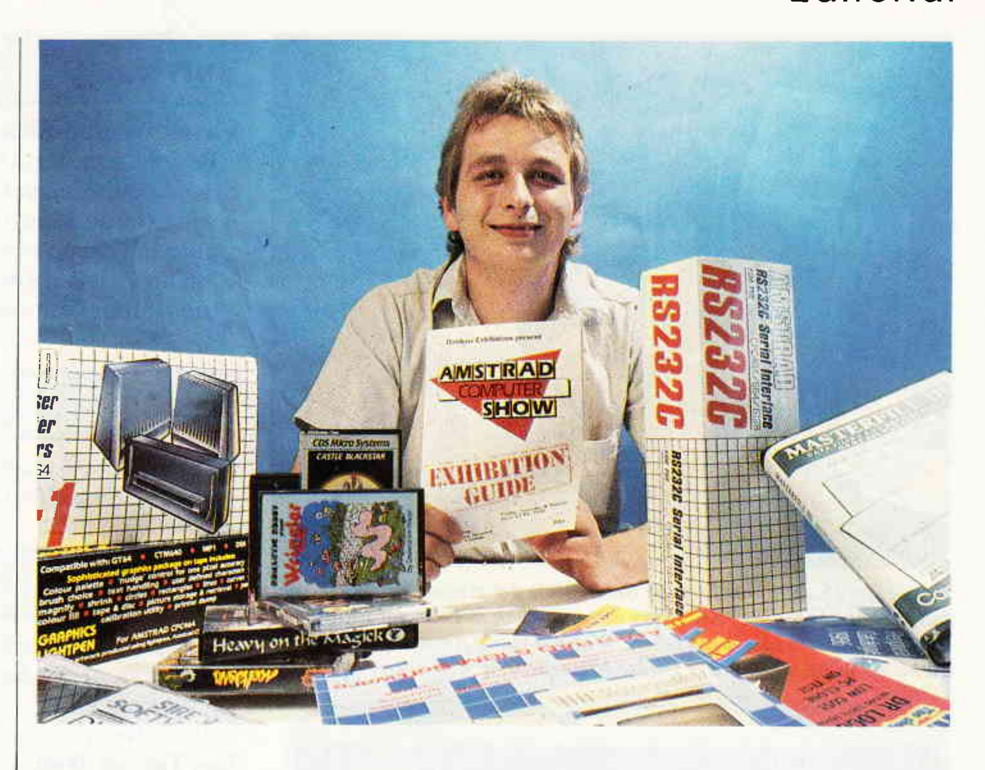
Druck
Druckerei Jungfer, 3420 Herzberg

Bezugspreise
"Schneider CPC International" erscheint monatlich am Ende des Vormonats.
Einzelpreis DM 6,-/sfr. 6,-/ÖS 50,-

Abonnementpreise
Die Preise verstehen sich grundsätzlich einschließlich Porto und Verpackung.
Inland:
Jahresbezugspreis: DM 60,-
Halbjahresbezugspreis: DM 30,-
Europäisches Ausland:
Jahresbezugspreis: DM 90,-
Halbjahresbezugspreis: DM 45,-
Außereuropäisches Ausland:
Jahresbezugspreis: DM 120,-
Halbjahresbezugspreis: DM 60,-
Bankverbindungen:
Postscheck Frankfurt/M: Kto.-Nr. 23043-608
Raiffeisenbank Eschwege:
BLZ: 522 603 85, Kto.-Nr. 245 7008
Die Abonnementbestellung kann innerhalb einer Woche nach Auftrag beim Verlag schriftlich widerrufen werden. Zur Wahrung der Frist reicht der Poststempel. Das Abonnement verlängert sich automatisch um 6 bzw. 12 Monate, wenn es nicht mindestens 6 Wochen vor Ablauf beim Verlag schriftlich gekündigt wird.
Für unverlangt eingesandte Manuskripte und Datenträger sowie Fotos übernimmt der Verlag keine Haftung.
Honorare nach Vereinbarung (die Zustimmung zum Abdruck wird vorausgesetzt).

Namentlich gekennzeichnete Fremdbeiträge geben nicht in jedem Fall die Meinung der Redaktion wieder.





Liebe Leser,

immer wieder hört man in der Computerbranche das Schlagwort "Sommerloch". Damit sind die heißesten Sommermonate gemeint, in denen viele Computer abgeschaltet bleiben und ein Sprung ins kühle Naß dem Computerspaß vorgezogen wird. Während dieser Zeit wird es meist ruhig auf dem Soft- und Hardwaremarkt, neue Produkte werden für die kalten Winterabende vorbereitet.
Die Situation auf dem Schneider/Amstrad-Markt stellt sich jedoch völlig gegensätzlich dar, hier tut sich mal wieder einiges.
Die vierte Amstrad-Show in England (ausführlicher Bericht im Innenteil) signalisierte ganz eindeutig: der Amstrad/Schneider-Markt kennt kein Sommerloch! Zur Zeit ist die ganze Branche am Rotieren, ständig sind Meldungen zu vernehmen, in denen neue Produkte angekündigt werden.
Der sagenumwobene, neue Schneider Computer ist zur Zeit allerdings noch nicht in Sicht; selbst in England hüllt man sich noch in großes Schweigen.
Der Besuch der Amstrad-Show war für die CPC-Redaktion trotzdem sehr informativ. Unser Bild zeigt nur einen kleinen Teil der neuen Produkte, die wir aus England mitgebracht haben.
Um Ihnen diese aktuellen Informationen nicht vorzuenthalten, mußten wir leider einige, bereits angekündigte Beiträge verschieben.
Davon betroffen ist zum einen der geplante HF-Modulator, den wir allerdings schon bald nachreichen werden. Zum anderen wurden der UDG-Designer und die zweite RS-232 Schnittstelle kurzfristig verschoben.
Wir meinen, daß diese kleine Veränderung der Aktualität von CPC International zugute kommt und wünschen Ihnen allen "heiße" Computertage bzw. einen erholsamen Urlaub.

Herzlichst Ihr

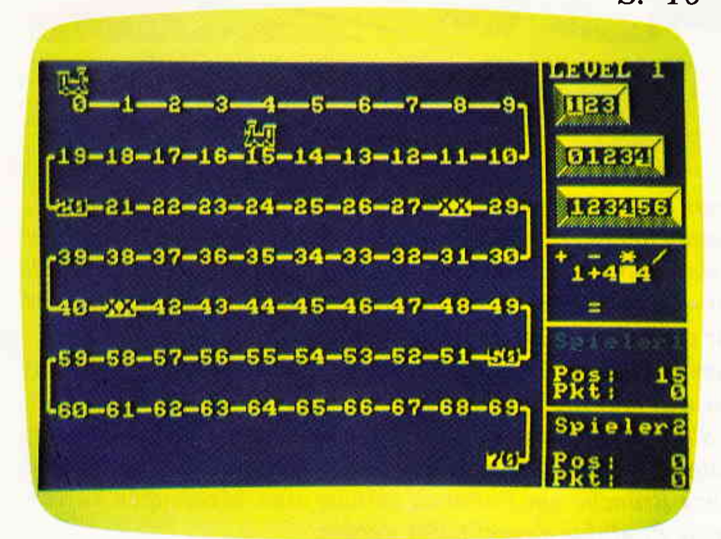
Stefan Ritter

(Stefan Ritter)



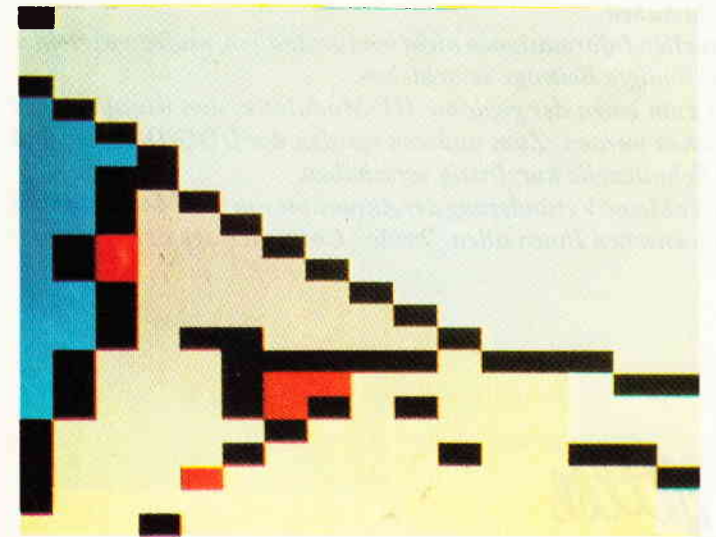
Vom 13. – 15. Juni war London Schauplatz der vierten Amstrad Computer Show. Ihre CPC-Redakion war natürlich mit von der Partie und hat sich aufmerksam umgesehen. Was die vierte Amstrad-Show alles zu bieten hatte, zeigt unser großer Messebericht.

S. 16



Lernen mit Hilfe des Computers – der Leitfaden unserer CAL-Rubrik. "Loco" ist ein pädagogisch hervorragend aufbereitetes Lernprogramm für Jung und Alt, das viel Spaß bereitet und zudem die Grundrechenarten spielerisch üben läßt.

S. 62



Mit unserem Programm-Hit des Monats aus der Sparte Anwendungen haben wir mal wieder einen Volltreffer gelandet. Das Grafik-Programm "Paint Master" erinnert in Bedienerkomfort, Geschwindigkeit und Leistungsfähigkeit an professionelle Grafiksysteme – überzeugen Sie sich selbst.

S. 112

Berichte:

Serie:

Hardware:

Tips & Tricks:

Joyce:

CP/M:

Programme:

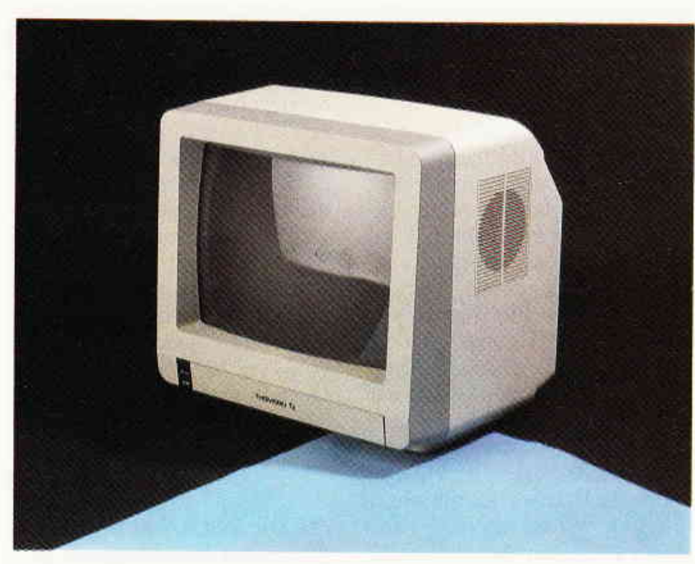
Lehrgänge:

CAL:

Software Reviews:

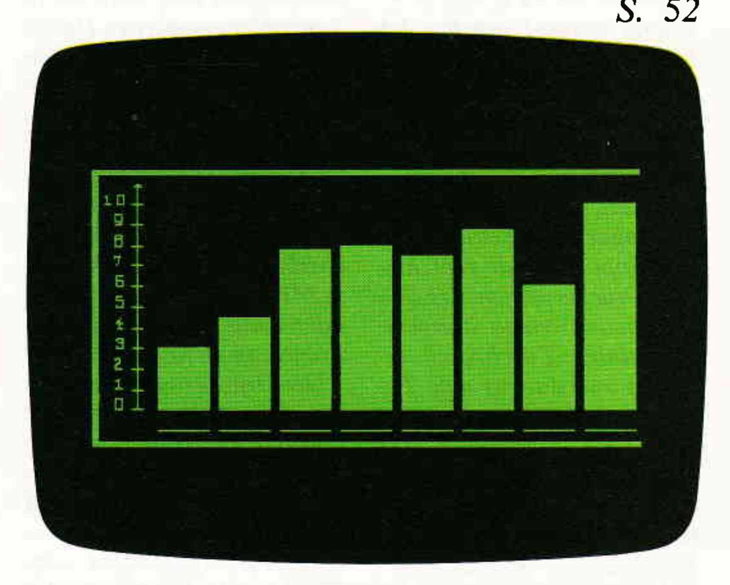
Abenteuer:

Rubriken:



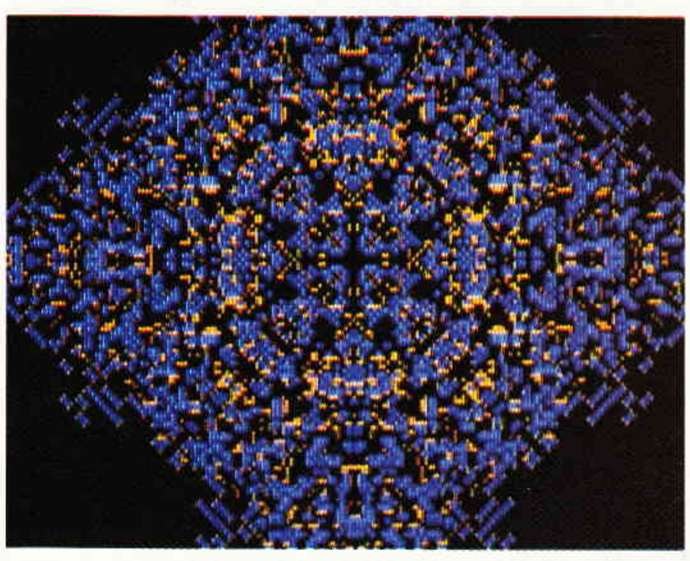
Der CPC geht fremd! Wir zeigen Ihnen, wie Sie externe Farbmonitore an Ihren CPC anschließen können. Wie immer, gibt es dazu jede Menge Grundlagen und Hintergrundinformationen – ein wahrer Leckerbissen!

S. 52



Joyce und Grafik – unter diesem Motto steht die Joyce-Rubrik. Mit einigen Tricks lassen sich auch auf dem Joyce Grafiken erzeugen, die auch ohne die berüchtigte GSX-Schnittstelle auskommen.

S. 91



Das Software-Experiment bietet diesmal eine originelle Version des bekannten Game of Life. Des weiteren erwartet Sie das Programm "Reproduktionsmuster" – eine in dieser Form wohl einmalige Simulation.

S. 120

Eine Bitte an unsere Leser

Die Rubrik »Leserbriefe« ist eine Einrichtung für alle Leser, die in irgendeiner Form Fragen, Probleme oder Anregungen zu Produkten, Programmierproblemen oder zu unserer Zeitschrift haben. Selbstverständlich sind wir bemüht, alle Leserfragen zu beantworten. Doch haben Sie bitte Verständnis, daß wir nicht alle eingehenden Briefe persönlich beantworten können. Oft erreichen uns mehrere Briefe zum gleichen Thema, einer davon wird dann stellvertretend für alle in unserer Zeitschrift beantwortet. Da auch wir nicht alle Fragen auf Anhieb beantworten können, müssen wir recherchieren. Und das dauert bekanntlich seine Zeit!! Wir möchten hiermit alle CPC-Leser noch einmal auf unseren Leser-Service hinweisen und bitten bei der Vielzahl der eingehenden Briefe um etwas Geduld. Für eilige Anfragen steht unsere Redaktion jeden Mittwoch von 17 - 20 Uhr am »Heißen Draht« zur Verfügung. Vielen Dank für Ihr Verständnis.

Ihre CPC-Redaktion

PS: Die Redaktion behält sich vor, Leserzuschriften in gekürzter Form wiederzugeben.

Context/Typenrad

Zuerst einmal möchte ich Sie herzlich zu dem Programm "Context" aus Heft 4/86 beglückwünschen. Es ist das erste Programm aus Programmzeitschriften, das auch das hält, was es verspricht und sogar ohne Fehler läuft. Zur Anpassung an verschiedene Druckerversionen (ESC-Sequenzen) werden Sie sicherlich einige Zuschriften erhalten und daraus eine Tabelle erstellen können.

Was aber, wenn man eine Druckerroutine benötigt, die keine dieser Sequenzen benötigt? Z.B. besitze ich eine Typenradschreibmaschine (Olympia-Carrera) mit Interface. Leider werden noch zwei ASCII-Zeichen anders als die Norm interpretiert (Zeichen "<" und ">"). Kurzentschlossen wird also eine Konvertierroutine geschrieben. Weil sie kurz und schnell sein soll, wird sie in Assembler angefertigt. Das so erstellte Programm wird in DATA-Zeilen abgelegt und in Context eingebaut. – Nur an welcher Speicheradresse darf sie abgelegt werden? Der normale Weg ist einfach mit dem MEMORY-Befehl den Speicherbereich festzulegen. Diese Methode endet aber früher oder später bei Context im Programmnirwana. Warum? – Der Druckertreiber arbeitet im Interrupt-Modus und schaltet Interrupts bei der Zeichenausgabe ab, und so kann zufällig RAM oder ROM (Adressbereich &0000 – &3FFF) eingeschaltet sein. Wo dann die Konvertierroutine unterbringen? Nun, das Maschinenprogramm disassemblieren und sehen, wo man es ändern kann, ist zu umständlich. Lediglich der CALL-Befehl zur Ausgabe eines Zeichens an den Centronic-Port erhält man so. (Adr.: &9C44). Eine andere Methode, den Textspeicher zu kürzen, gefällt mir nicht. Auf der Suche nach weiteren Lösungen fiel mir der SYMBOL AFTER-Befehl auf. Hierbei wird eine Zeichenmatrix im RAM untergebracht, auch im Programm Context. Es werden jedoch die Zeichen 127 bis 199 gar nicht zur Darstellung benötigt. Folglich ist es ohne Bedeutung, wie diese Zeichenmatrix aussieht. Ab Adresse &A1D3 beginnt dieser Zeichensatz in Context. Acht Byte pro Zeichen ergeben die Adresse &A2F4 für das Zeichen 127. Hier kann also das Konvertierprogramm beginnen. Außerdem gehört dieser Speicherbereich zum nicht überlappten RAM-Bereich. Dadurch ist ein Störung durch ROM/RAM-Schaltungen ausgeschlossen.

Die Konvertierroutine fängt nun alle ESC-Sequenzen ab und kodiert die Zeichen "<" und ">" in die benötigten Zeichenfolgen um. Dabei hilft die Tatsache, daß jede ESC-Sequenz aus drei Zeichen besteht. Unterstreichen ist mit einer Schreibmaschine durch Zurücksetzen des Schreibkopfes möglich. Eine einfache ESC-Sequenz reicht dazu nicht. Deshalb legt die MC-Routine noch einen eigenen Merkspeicher (3 Byte) an und unterstreicht jedes Zeichen, bis die Sequenz zum Abschalten (DB 27,45,0) erscheint. Um das Programm Context bedienerfreundlich zu halten, werden noch einige Schönheitsänderungen neben der Konvertierroutine ins Programm eingefügt.

Die Anderungen sehen dann folgendermaßen aus:
(siehe Listing Context/Typenrad)

Reinhard Pekx, Duisburg

Listing Context/Typenrad

```
  85 conv=&A2F4
 185 GOSUB 2235
1340 nlq=1:CALL conv:GOSUB 1830:GOTO 1290
1350 nlq=0:POKE &9C44,&31:POKE &9C45,&BD:GOSUB 1830:GOTO 1290
1830 PEN 1+nlq:LOCATE 16,4:PRINT"Olympia"
1840 PEN 2-nlq:LOCATE 16,6:PRINT"EPSON":RETURN
2235 RESTORE 2270:x=0:FOR i=0 TO 133:READ j:x=x+j
2240 POKE conv+i,j:NEXT i:READ j
2245 IF x=j THEN RESTORE:RETURN ELSE 2410
2270 DATA &21,&05,&A3,&22,&44,&9C,&AF,&32,&7A,&A3
2280 DATA &32,&7B,&A3,&32,&7C,&A3,&C9,&F5,&3A,&7A
2290 DATA &A3,&B7,&28,&12,&3D,&32,&7A,&A3,&28,&06
2300 DATA &F1,&32,&7C,&A3,&37,&C9,&F1,&32,&7B,&A3
2310 DATA &37,&C9,&F1,&FE,&1B,&20,&07,&3E,&02,&32
2320 DATA &7A,&A3,&37,&C9,&E5,&D5,&2A,&7B,&A3,&11
2330 DATA &01,&2D,&A7,&ED,&52,&D1,&E1,&20,&0C,&F5
2340 DATA &3E,&5F,&CD,&64,&A3,&3E,&08,&CD,&64,&A3
2350 DATA &F1,&F5,&FE,&3C,&28,&11,&FE,&3E,&20,&07
2360 DATA &3E,&1B,&CD,&64,&A3,&3E,&64,&CD,&64,&A3
2370 DATA &F1,&37,&C9,&3E,&1B,&CD,&64,&A3,&3E,&62
2380 DATA &18,&F1,&CD,&2E,&BD,&38,&FB,&C5,&06,&EF
2390 DATA &E6,&7F,&ED,&79,&F6,&80,&ED,&79,&E6,&7F
2400 DATA &ED,&79,&C1,&C9, 16825
2410 PRINT"Fehler in DATAs !!!":STOP
```

Context – Okimate 20

Vorab will ich Ihnen erst mal sagen, daß dieses Textverarbeitungsprogramm aus meiner Sicht wirklich spitze ist. Besonders toll ist, daß sich die Mühe des Abtippens, Fehler-(Tipp- usw.) ausmärzens und der Druckeranpassung absolut gelohnt hat. Machen Sie weiter so!

Nun zur Anpassung:
Der Okimate 20 bietet über die im Context-Programm gebotenen Möglichkeiten noch die Umstellung auf Elite, Kursiv und die Wahl zwischen diversen nationalen Buchstabentypen. Gleich vorweg: dieses zu realisieren ist mir nicht gelungen.

Anschließend einige Programmänderungen (s. Tabelle Programm-Änderungen):
Die weiteren Änderungen beziehen sich ausschließlich auf den Maschinencode.

Fettdruck:
AUS
4500 – &49 an &9F2A (3. Byte)
EIN
4530 – &54 an &9F42 (3. Byte)

Unterstreichen:
AUS
4480 – &44 an &9F1E (7. Byte)
EIN
4510 – &43 an &9F33 (4. Byte)
4510 – 00 an &9F34 (5. Byte)

Zeichenverdichtung (condensed):
AUS
4490 – &1E an &9F23 (4. Byte)
(1E entspricht Rückkehr zu PICA)
EIN
4520 – &1D an &9F38 (1. Byte)

Vergrößerte Schrift (gedehnt):
AUS
4490 – &1E an &9F20 (1. Byte)
(1E entspricht Rückkehr zu PICA)
4490 – 00 an &9F21 (2. Byte)
EIN
4510 – &1F an &9F35 (6. Byte)
4510 – 00 an &9F36 (7. Byte)
4510 – 00 an &9F37 (8. Byte)

Papiermangel nicht übergehen:
4470 – 00 an &9F11 (2. Byte)
4470 – 00 an &9F12 (3. Byte)

Tiefgestellte Zeichen (Index):
EIN
4520 – &4C an &9F3F (8. Byte)
4530 – 00 an &9F40 (1. Byte)
AUS
4490 – &4D an &9F27 (8. Byte)

Hochgestellte Zeichen (Exponent):
EIN
4520 – &4A an &9F3C (5. Byte)
AUS
Über zweimaliges Hinzufügen des Index-Zeichens, ansonsten zu handhaben wie in der Programmbeschreibung dargestellt.

Programm - Änderungen

```
       Basic                                                MC
NLQ
   80 - nlq=&9F1B                                 4480 - &31 an &9F1B (4.Byte)
 1340 - POKE nlq,&31                              4480 -  00 an &9F1C (5.Byte)
 1350 - POKE nlq,&30
 1830 - PEN 1+(PEEK(nlq)-48)
 1840 - PEN 2- PEEK(nlq)+48

Zeilenabstand
 1380 - an=PEEK(mzei)+1:IF an>101 then             170 - &3C an &9322 (3.Byte)
        an=21                                      120 - 11198
 1390 - POKE mzei,an:POKE cabs,1440/(an-1)        4470 - &1B an &9F16 (7.Byte)
 1850 - an=PEEK (mzei)
 1860 - PEN 2:LOCATE 20,12:PRINT USING            4470 - &25 an &9F17 (8.Byte)
        "##.#";(254/(an-1))-2.4                   4480 - &39 an &9F18 (1.Byte)
 1870 - :LOCATE 20,14:PRINT an:RETURN             4480 - &3C an &9F19 (2.Byte)

 an=max.Zeilenzahl/Seite.Bei Start ist der Standard 60 Zeilen.
```

wenden. Ich wäre Ihnen dankbar, wenn Sie mir bei meinem Problem helfen könnten.
Und hier die Anpassung des Programms an den Star NL-10:
Änderungen im CONTEXT-MC-Basiclader
Basiczeilen 4440 und 4470 ändern in:
4440 DATA 6123
4470 DATA 00,1B,38,00, 1B,52,02,1B
Änderungen im Hauptprogramm:
Basiczeilen 80, 1360, 1890 und 1920 ändern in:
80 casin=&9E02:cata= &9E3E:cset=&9F16: cabs=&9F19:nlq= &9F1C
1360 IF PEEK(cset)=2 THEN POKE cset,0 ELSE POKE cset,2
1890 IF PEEK(cset)=0 THEN PRINT"I" ELSE PRINT"D"
1920 IF PEEK(cset)=0 THEN 2050

Lothar Flegel, Willroth

CPC:
Wir geben Ihre Frage an unsere Leser weiter. Wer einen NL-10 besitzt und das Problem gelöst hat, soll uns schreiben.

Der Grund hierfür liegt darin, daß beim Okimate 20 das hoch- bzw. tiefgestellte Zeichen separat abgeschaltet wird und im Programm dafür kein Platz ist.

Rainer Beinlich, Braunschweig

Context Star SG 10

Als erstes möchte ich mich bei Ihnen für dieses hervorragende Programm bedanken, mit dem auch dieser Brief geschrieben ist.
Doch nun zu den Änderungen:
Für die Steuersequenzen sind folgende Speicherstellen umzupoken.
POKE &9F15,55
POKE &9F16,n
POKE &9F1B,66
POKE &9F23,27
POKE &9F24,66
POKE &9F25,1
POKE &9F30,66
POKE &9F31,5

Außerdem müssen im Basicprogramm folgende Zeilen geändert werden:
80 :cset=&9F16:
1340 POKE nlq,4:
1350 POKE nlq,5:
1360 if PEEK(cset)=0 then POKE cset,2 else POKE cset,0
1890 if PEEK(cset)=0 then print"I" else print "D"
1920 if PEEK(cset)=0 then 2050

Da der Star SG-10 im Star-Modus keinen /216 Zeilenvorschub besitzt, muß in Zeile 1390 der Wert :POKE cabs, 2304/n: geändert werden in :POKE cabs,1536/n:

Horst Kufferath, Duisburg

Filecopy

Hiermit übersende ich Ihnen einen, meines Erachtens, sehr nützlichen Tip, der die Benutzer einer Vortex RAM-Erweiterung anspricht. Wie Ihnen ja sicherlich bekannt ist, kann zum Kopieren der Programme von Laufwerk A nach C, außer schwer erhältlichen diversen Programmen, nur das auf der jeweiligen CP/M-Systemdiskette mitgelieferte Programm "PIP" benutzt werden. Hier muß allerdings mit "WILDCARDS" auf sehr umständliche Art und Weise kopiert werden. Abhilfe schafft hier das von der Firma Vortex zur Speichererweiterung mitgelieferte Programm "FILECOPY.COM", wo jedoch noch kleinere Veränderungen vorgenommen werden müssen (Filecopy V2.6).
Dieses läuft wie folgt ab:
Laden Sie das auf der Systemdiskette befindliche Programm "DDT.COM" mit dem eben erwähnten "FILECOPY.COM" in den Speicher Ihres CPC's.
DDT FILECOPY.COM
Hiernach meldet sich das Programm "DDT.COM"
DDT Vers 2.2
Next PC
0C00 0100
Geben Sie folgendes ein:
S022F
Nun erscheint die zu ändernde Speicherstelle samt Inhalt auf dem Bildschirm, gefolgt von einem Cursor, wo Sie den neuen Wert hineinschreiben können. Hier geben Sie nun anstelle der 42 eine 43 ein.
022F 42 43
Ebenso verfahren Sie mit den Speicherstellen
0247 0906 092b
Beenden Sie nun das Programm "DDT" mit dem Befehl G0'(null)'.
Ist dies geschehen, so geben Sie
SAVE 11 CCOPY.COM
in Ihren Rechner ein, womit dann ein CP/M-File auf Ihrer Diskette entstanden ist, mit dem Sie sowohl selektiv als auch total von Laufwerk A nach C und umgekehrt kopieren können.

Guido Weise, Münster

Context Star NL-10

Kürzlich habe ich mir den Drucker Star NL-10 zugelegt. Sogleich ging ich an die Arbeit und tippte Ihr Textverarbeitungsprogramm Context ein. Als einzigste Anpassung an den Star NL-10 mußte ich Änderungen für die Einstellung des deutschen Zeichensatzes vornehmen. Sonst läuft Context hervorragend und ich bin sehr zufrieden mit diesem Programm. Eine Frage hätte ich allerdings noch. Wie kann ich andere Steuercodes (z.B. Kursivschrift) an den Drucker senden. Allerdings möchte ich dafür keine schon belegten Tasten verwenden. (Fortsetzung oben)

Context - DMP 2000

Zur Anpassung an den Drucker DMP 2000 müssen im Programm folgende Änderungen vorgenommen werden:
Im DATA-Block 25 des Basicladers:
Zeile 4440 DATA 6123
Zeile 4470 DATA 00,1B,38, 00,1B,52,02,1B
oder:
POKE &9F15,82 (bzw. &52):POKE &9F16,02
Im Hauptprogramm müssen folgende Änderungen vorgenommen werden:
In Zeile 80 cset=&9F15 ändern auf cset=&9F16
In Zeile 1360 die Zahlen 54 und 55 ändern auf 02 und 00
In Zeile 1790 die Zahl 55 ändern auf 00
In Zeile 1920 die Zahl 55 ändern auf 00

Willfried Vandrey, Stuvenborn

Knack die Nuß

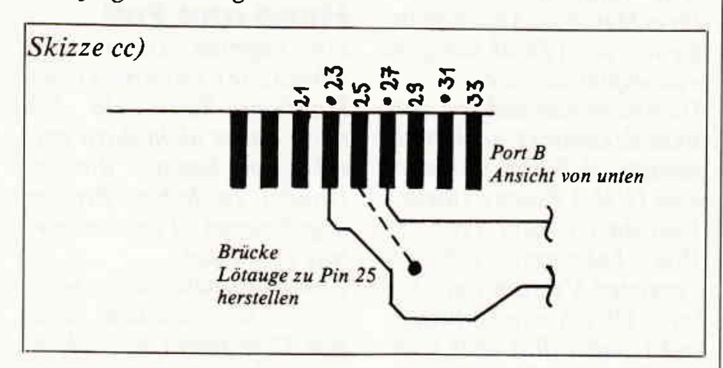
Im Beitrag "Knack die Nuß", Heft 6/86 hat sich ein Fehler eingeschlichen. Beim Ausdruck des Listings 1 ist uns in Zeile 170 ein Fehler unterlaufen, der zum Abbruch des Programmes führt. Richtig heißt es: X = a/b/c/d/e/f/g.

5 1/4" am CPC 464/664

So muß sich der CPC melden, wenn alles korrekt abläuft:
|cpm

a>COPYDISC 2.0
Please insert source disc into drive A and destination disc into drive B then press any key:__
Copying started
Reading track 1 ((..8) usw. (bis 40))
Writing track 1 ((..8) usw. (bis 40))
Copying complete
Do you want to copy another disc(Y/N):__

Ich benutze das Laufwerk BASF 6128 Nr. 81670-001 schon mehrere Monate ohne irgendwelche Beanstandungen. Das gleiche gilt auch für das Tandon Laufwerk.
Beide Laufwerke sind Doppellaufwerke, die allerdings nur einseitig genutzt werden, da der Controller leider einiger Funktionen beraubt ist. Das Ansprechen der Laufwerksmotoren einschließlich Relais ist normal und liegt an der von Schneider benutzten Dekodierung. Hier wäre noch eine Verbesserung möglich, da das Geklicker des Relais unter Umständen lästig ist.
Die von Ihnen angegebene Fehlermeldung läßt auf eine Unterbrechung im Kabel schließen bzw. fehlt an Ihrem CPC die Verbindung auf der Unterseite des CPC 664 vom Lötauge zur Leiterbahn 25 des Port B. Dieser Portanschluß ist dafür zuständig, das Laufwerk B zu selektieren (siehe hierzu Anschlußbild Pin 12 des BASF 6128). Warum am Port B die Verbindung zu Pin 25 fehlt, ist mir ein Rätsel, denn wie soll zum Beispiel das Zweitlaufwerk auch von Schneider ohne diese Verbindung laufen?
Bei der Reproduktion der Skizze ist eine kleine Ungenauigkeit aufgetreten, die ich nachfolgend korrigiere:



Wenn Sie die Umbauanleitung genau einhalten, läuft das Zweitlaufwerk auf Anhieb. Ich gehe davon aus, daß das Laufwerk wirklich in Ordnung ist. Mir wurde z.B. auf dem Flohmarkt ein BASF 6106 verkauft, welches trotz ausdrücklicher Versicherung nicht in Ordnung war.
Nun viel Erfolg beim Anschluß.

Hans Ziegler, Hundsangen

Autostart-Fehler behoben

Sicher hat sich schon jeder Maschinensprache-Programmierer über den "Fehler" im Betriebssystem des CPC 464 mit Floppy geärgert. Wird nämlich ein mit SAVE "NAME",b,Start,Länge,Einsprung gespeichertes Programm mit RUN"NAME" gestartet, wird die Floppy abgeschaltet, so daß man diese nicht weiter benutzen kann. Die Floppy wird abgeschaltet, da bei dem Befehl RUN"NAME" vor dem Start des Programmes, die Routine MC START PROGRAMM angesprungen wird. Diese Routine initialisiert nun alle Packs neu, unter anderem auch die Kassettenroutinen.
Dies führt dazu, daß man die Programme mit:
10 memory xxxx
20 load "Name"
30 call xxxx
startet. Dies kostet jedoch jedesmal 1K Speicherplatz auf der Diskette. Mit einigen Befehlen kann man das Problem lösen. Die Floppy wird wieder angeschaltet, wenn man am ANFANG des Programmes folgende Befehle einfügt:
C5 ; PUSH BC
E1 ; POP HL
OE 07 ; LD C,&07
CD CE BC ; CALL &BCCE
Ist eine Reservierung des Speichers durch den MEMORY-Befehl notwendig, kann man das mit:
21 XX XX ; LD HL, Memory Wert
22 7B AE ; LD (&AE7B),HL
erreichen.

Oliver Kocks, Hürth

Hand und Fuß – Joyce berücksichtigen?

Ich freue mich sehr, daß Sie in Ihrer Zeitschrift auch an Joyce-Benutzer denken.
Bitte machen Sie weiter mit Tips und Programmen für dieses Gerät.
Können Sie auch in der neuen Serie "Hand und Fuß" den Joyce berücksichtigen? Das wäre sehr gut, denn in der Hinsicht gibt es bisher keine Veröffentlichungen.
Wenn das zeitlich nicht mehr geht, könnten Sie vielleicht am Ende der Serie einige Tips geben, wie man die Schaltungen für den Joyce abändern kann?

Bernd Franck, Oldenburg

CPC:
Die Schnittstelle zwischen Joyce und SCHNEIDER-WARE ist in Arbeit. Mit einem Adapterbaustein wird dann den Joyce-Besitzern die CPC-Hardware zur Verfügung stehen.

Context-Copy

Nachdem mir das Abtippen des Programmhit des Monats "CONTEXT" doch zu mühselig wurde, habe ich mich Ihres Databox-Angebots bedient. Vielen Dank übrigens – prompter Service.
Dummerweise hab ich zum Bestellzeitpunkt noch nicht gewußt, daß ich mir doch eine DDI-1 Floppy zulege.
Und da ist mein Problem. Wie bekomme ich die Cassetten-Version auf Diskette. Ob ich nun Context1. und Context.Bin oder Context.Bin und Context oder sonstige Kombinationen von Tape auf Disc nehme, nach Einladen von Diskette wird mir die Zeile 100 des Hauptprogramms immer als "Improper argument" gemeldet.
Rainer W. Schulz
Wilhelmshaven

CPC:
ITAPE.IN [Enter]
Memory & 92FF [Enter]
Load "Context.Bin" [Enter]
Save "Context.Bin,B,&9300, &D00 [Enter]
... und schon ist Context auf Disk!

Hand und Fuß – sehr interessant

Da es mittlerweile auf dem Meßgerätemarkt einige hundert Meßgeräte gibt, die über den genormten IEC-Bus-Anschluß verfügen, würde ich es für sinnvoll halten, wenn im Rahmen Ihrer Hardwareserie ein entsprechendes Interface einschließlich der zugehörigen Treibersoftware vorgestellt würde.
Damit könnte sich den CPCs mancher meß- und steuerungstechnische Einsatz eröffnen.
Da ich selbst solche Anwendungen hätte, würde ich mich freuen, wenn Sie mir mitteilen, ob mit solcher Hard- und Software zu rechnen ist.
Erich Kopp
St. Georgen

CPC:
Die Veröffentlichung einer IEC-Karte ist für den weiteren Verlauf der SCHNEIDERWARE geplant. Vorläufig sollen allerdings erst einige Grundkarten angeboten werden.

Hand und Fuß

Ich begrüße Ihren Entschluß zur Einführung einer Hardware-Serie, da ich mich bisher nicht dazu entschließen konnte, die im Handel zu hohen Preisen angebotenen Peripheriegeräte zu kaufen.
Zum Aufbau der Serie möchte ich Sie bitten, keine von Computern gezeichneten Schaltpläne zu veröffentlichen (abschreckendes Beispiel auf S. 72 im Mai-Heft), jeweils ein Foto des Fertiggerätes abzubilden und zu jedem Gerät eine Bauteileliste abzudrucken.
Zur Software, die für die einzelnen Bausätze nötig ist, möchte ich Sie bitten, die Möglichkeit offenzuhalten, daß der Anwender die Bildschirmausgabe selbst gestalten kann, d.h. diesen Programmteil in BASIC zu schreiben.
Jürgen Münch
Triefenstein

CPC:
Zum Thema Schaltplan sind wir noch auf der Suche nach einer geeigneten Lösung. Die Software zu den Bauanleitungen wird natürlich universell gestaltet, daß jeder Anwender zufriedengestellt wird.

Unbekanntes Zeichen

Ich besitze seit September 1985 einen Schneider CPC 464 Computer, und wende mich heute mit folgendem Problem an Sie. In mehreren Programmlistings verwenden Sie das Zeichen chr$(160). So auch in Ihrer Ausgabe Sonderheft 1/86 bei dem Programm "Wallstreet" in Zeile 770 und 780. Ich möchte nun wissen, wie ich dieses Zeichen über die Tastatur eingeben kann und welche Bedeutung es hat? Ich habe sogar versucht, das Zeichen über eine Sondertastenbelegung einzugeben, beim anschließenden Listen war das Zeichen jedoch weg. Vielleicht können Sie mir sagen, woran das liegt?
Michael Suntrup
Nordhorn

CPC:
Das kleine Dach ist das Potenzierungszeichen des Lokomotive-Basics. Sie erreichen es über die Pfund/Pfeil-Taste in der oberen Reihe der Tastatur.

Fehler suchen und finden

Das Programm "Suchen und Finden", aus Heft 3/86, enthält in den Zeilen 300 und 390 einen Druckfehler:
Zeile 300 muß heißen:
300 PZ$(I)=LEFT$(PZ$(I),SP-1)+ERS$+MID$(PZ$(I),SP+LA)
und Zeile 390 muß heißen:
390 FOR I=0 TO K: PRINT#9,PZ$(I):NEXT

Hans Schwertner
Jülich

Programm gerettet

Üblicherweise ist ein Programm verloren, wenn die Meldung "Read Error.B" erscheint und trotz mehrfachem Hin- und Herspulen des Bands die Programmaufnahme nicht funktioniert.
Ziemlich einfach läßt sich das Programm trotzdem retten, wenn man dafür zwei normale Tonbandgeräte einsetzt und das Programm ganz normal so überträgt, wie sonst ein Musikstück. Man kann also seine Stereoanlage zusammen mit einem einfachen Tonbandgerät, über das Radio gekoppelt, einsetzen. Das Programm wird nun einfach überspielt und siehe da, bei der Neuaufnahme ist der Fehler "Read error b" verschwunden. Meistens klappt das sofort, ansonsten noch einmal wiederholen.

Ich hoffe, daß dieser Tip so manchem User behilflich sein kann.
Henry Dähn
Bad Harzburg 1

Super Saturn

Im o.g. Listing aus Sonderheft 1 sind einige – wenn auch unbedeutende – Fehler enthalten. Der Autor hat sich jedoch so viel Mühe damit gegeben, einen möglichst "naturgetreuen" Sound beim Ablauf des Spiels zu erzeugen, daß es einfach zu schade ist, diesen wegen einiger Flüchtigkeitsfehler zu unterschlagen.
Will man auch akustisch in den vollen Genuß des Spieles kommen, dann müssen folgende Zeilen geändert werden:
2080 GOSUB 3100
2310 GOSUB 3150
3110 FOR n=4 TO 1 STEP-1

Harry Liedtke, Hückelhoven

Zahl in String

Ihre Antwort auf den Leserbrief mit dieser Überschrift in Heft 6/86 ist nur teilweise richtig, ein kleiner Fehler hat sich eingeschlichen. Sie schreiben:
A=5:A$=STR$(A)
und schreiben jetzt in A$ steht "5".
Das ist falsch! In A$ steht tatsächlich "CHR$(32)5!!
Es ist zwar richtig, daß mit STR$ Zahlen in Strings umgewandelt werden, aber in dem Moment, in dem Sie einen Vergleich anstellen, erhalten Sie eine negative Antwort! Siehe Listing 1
Richtig wäre folgende Umwandlung = A$=MID$ (STR$(A),2)
Ändern Sie dazu Zeile 11 in Listing 1.

Bernd Uhlmann, Aachen

```
10 A=5
11 A$=STR$(A): REM *** RICHTIGE ZEILE 11 A$=MID$(STR$(A),2) ***
12 PRINT LEN(A$)
13 AA$="5"
14 PRINT LEN(AA$)
15 PRINT "1.STELLE VON A$ IST CHR$(";
16 PRINT ASC(LEFT$(A$,1));")"
17 IF AA$=A$ THEN PRINT "GUT":GOTO 19
18 PRINT "SCHLECHT, DA TEXTINHALTE VON A$ UND AA$ NICHT GLEICH"
19 END
```

Listing 1

Logon Please!

Die in der Juni-Ausgabe Ihrer Zeitschrift angesprochene Kennwort-Angabe ohne Bildschirm-Echo läßt sich auch ein wenig einfacher gestalten:
*100 PRINT"LOGON PLEASE ...": **CALL &BB57** 'Bildschirm-Echo aus*
110 INPUT" ",password$
120 IF password$="XYZ" THEN 130 ELSE 100
130 **CALL &BB54** *'Bildschirm-Echo wieder einschalten*
140 'Programm ausführen ...

Andreas Dippe, Neunkirchen

Text von Floppy

Wieder durften wir, die CPC International-Leser, ein wunderbares Programm bzw. Erweiterung zum bestehenden CONTEXT-Programm, aus Ihrem Heft (5/86) entnehmen. Als ich die gelungene Erweiterung auf dem Bildschirm sah, fiel mir sofort das Suchen beim Laden von Texten auf. So dachte ich mir, könnte man die Erweiterung auch zum Laden der Texte benutzen, um den gesuchten Text direkt vom Bildschirm übernehmen zu können. Die neue Variable **la** *rem:y bekommt durch das Unterprogramm ab Zeile 541 einen neuen Wert*
neue Programmzeilen:
Die Zeile 559 wird erweitert:
559 **if la=1 then return else** *window swap 0,1*

625 if y=6 then la=l:gosub 541
820 if la=l then 880

Die **alte** *Zeile 820 wird 825*
825 if y=10 then 940

Zeile 900 ändern:
900 if **la=1** *then call cltxt, zadr,az*80:call casin, zadr,az, §na$*

Zeile 920 erweitern:
920 for i=l to 2000:next: window swap 0,1: **if la=1 then la=0:goto 185 else** *1000*

Jedoch fiel mir bei der cat-Routine auf, daß, wenn die Diskette volle 64 Eintragungen hat, das Window überschrieben wird. Falls ein User dazu eine Lösung hat, bitte ich ihn, diese Ihrer Zeitschrift mitzuteilen, damit alle User ihre Programme verbessern können.

Peter Wiesemann, Haan

Hardcopy auf dem Star SG 10

Herr Altfeld hat uns mit seiner DIN-A4-Hardcopy (CPC 5/86, S. 80) aus der Patsche geholfen, ohne es zu wissen.

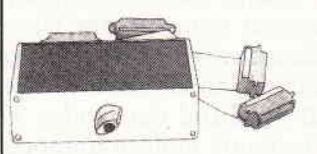
Alle Hardcopy-Programme, die wir bisher auf unserem Star SG 10 drucken lassen wollten, scheiterten am falschen (zu großen) Zeilenvorschub. Eine winzige Änderung in Herrn Altfelds "Supercopy" brachte nun die Lösung:

630 DATA C8,9C,3E,08, CD,C0,9C,3E, 1049

Die Änderung &08 (dez. 8) anstatt &0C (dez. 12) ergibt den Zeilenvorschub 8/144, also 4/72 (Drucker-Befehl ESC "3" 8).
Na klar – die von dem Programm erzeugte Druckzeile ist eben vier Punkte hoch!
Mit der Änderung dieses Drucker-Befehls müßte es übrigens auch gelingen, anderen Hardcopy-Versionen auf dem Star-Drucker Beine zu machen.

Thomas Lipka, Christoph Lipka, Martin Lipka, Plettenberg



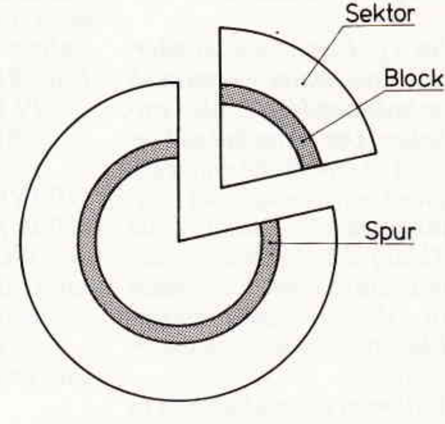


Floppykurs Teil 2

Im ersten Teil unseres Floppykurses haben wir uns mit dem grundsätzlichen Aufbau einer Floppy-Hardware auseinandergesetzt.
Warum die Hardware so und nicht anders aufgebaut sein muß, wird Ihnen allerdings erst klar, wenn Sie etwas über das Grundprinzip und den eigentlichen Aufbau einer Diskette erfahren haben.

1.Grundlegendes über Datensicherung (Magnetismus und seine Folgen)

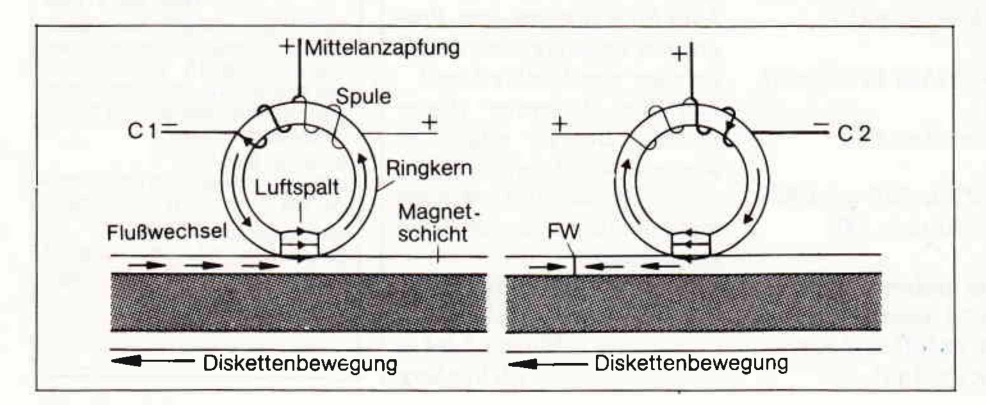
Eine magnetische Datenaufzeichnung funktioniert nach dem gleichen Prinzip, mit dem Sie eine Audioaufnahme auf Ihrem Heimkassettenrecorder vornehmen. Auf einem Trägermaterial aufgebrachte, stabförmige magnetische Teilchen werden beim Bespielen von einem Elektromagneten in eine bestimmte Anordnung gebracht (siehe ABB.1).
Beim Wiederabspielen erzeugen diese Magnetteilchen in der Spule des Tonkopfes eine induktive Spannung, die von der Recorderelektronik in akustische Signale (oder Daten) umgesetzt werden. Deutlich wird dies, wenn Sie eine Datenkassette mit einem normalen Tape-Deck anhören.
Der Datenträger der Diskette besteht aus dem gleichen Material wie eine Kassette, und auch die Daten werden durch einfaches Aufnehmen und Wiederabspielen auf die Diskette bzw. in den Rechner gebracht.
Bei einer Kassette liegt der Nachteil im mechanischen Aufbau der Lesevorrichtung. Dadurch, daß das Trägermaterial nur in einer Richtung am Tonkopf vorbeigeführt werden kann, kommt man um ein Umspulen der Kassette nicht herum. Einzelne Stellen auf einem Band wiederzufinden, ist aufgrund der labilen Zählwerksmechanik nahezu unmöglich.
Eine Diskette müssen Sie sich nun so vorstellen, daß das Bandmaterial kreisförmig um die Mittelachse angeordnet ist. Dadurch, daß die Diskette um die Mittelachse rotiert (siehe Teil 1), sind die Entfernungen, die der Tonkopf zurücklegen muß, um etwas zu finden, schon wesentlich geringer. Zudem rotiert die Diskette wesentlich schneller, als sich ein Tape dreht. Dies bringt Vorteile in Bezug auf Geschwindigkeit und Speicherkapazität. Damit wäre schon einmal erklärt, warum sich eine Diskette dreht. Bleibt aber immer noch die Frage offen, wieso der Tonkopf auch noch eine vertikale Bewegung ausführen muß.
Eine Kassette hat ein ungefähr 0,5 cm breites Band. Durch die hohe Geschwindigkeit, mit der der Datenträger der Floppy am Tonkopf vorbeigeführt wird, lassen sich auf dünnere Magnetschichten höhere Datenmengen bringen. Warum das so ist, ist uninteressant. Würde das Band genauso schnell am Tonkopf vorbeilaufen wie die Floppy, so würde ein 10 mm breites Tape vollständig ausreichen. Da eine Diskette aber einen Radius von 2,5 cm hat, ließen sich auf ihr 40 Bandstücke von 10 mm Dicke unterbringen. Alle kreisförmig um die Mittelachse angeordnet. Beim Vortexformat paßten sogar 80 dieser Bandstücke auf eine Diskette (logisch, die sind ja auch größer). Der Steppermotor 2 kann den Tonkopf (Schreib-/Lese-Kopf) nun in 40 Schritten à 10 mm vor- oder zurückbewegen, und ist somit in der Lage, zwischen den einzelnen Bandstücken hin und her zu wechseln. Selbstverständlich klebt kein Floppyhersteller Bandstücke auf seine Datenträger. Die Disketten kommen mit einer magnetischen Beschichtung und die Bandstücke, die wir im weiteren jetzt SPUREN (Tracks) nennen, müssen vom Anwender selbst auf die Diskette FORMATIERT werden.
Soviel zu den Prinzipien der magnetischen Aufzeichnung. Alles, was über das Gesagte hinausgeht, fällt in den Bereich der Grundlagenforschung



und ist nicht so interessant. Der vorhergehende Absatz endete mit dem schönen Wort FORMATIEREN und bietet uns einen idealen Einstieg in die Architektur der Diskette.

2. Ordentlich wie ein Regal (Sektoren und Tracks)

Formatiert werden muß jede Diskette, bevor man sie benutzen kann. Der Anwender leitet mit dem Formatieren einen wahnsinnig komplizierten Vorgang ein, in dem die einzelnen Spuren vormagnetisiert werden und noch eine zusätzliche Unterteilung in 9 Sektoren pro Spur bekommen. Jeder dieser Sektoren kann eine Datenmenge von 512 Byte halten.
Je nachdem, welches Format vom Anwender gewählt wurde, werden bestimmte Spuren zur Aufnahme von Floppy-internen Routinen reserviert. Den Unterschied zwischen den einzelnen Formaten erläutern wir zu einem späteren Zeitpunkt, wenn es wirklich ans Eingemachte geht.
Nach dem Formatieren, welches über ein Programm Ihrer Masterdiskette vorgenommen werden kann, befinden sich auf einer Diskettenseite auf jeden Fall 40 Spuren à 9 Sektoren (beim Vortexformat 80 Spuren à 9 Sektoren).
Um nun auf der Diskette ein Programm zu finden, welches auf Spur 5, Sektor 6 steht, muß der Schreib-/Lesekopf zunächst einmal in die Ausgangsstellung Spur 0, Sektor 0 gebracht werden, danach der Steppermotor 1, die Diskette sechs Schritte um die Mittelachse bewegen und Steppermotor 2 fünf Schritte in Richtung auf die Mittelachse fahren.
Danach befindet sich der Schreib/Lesekopf genau über Spur 5, Sektor 6 und kann die 512 Bytes, die in diesem Sektor stehen, lesen. Trat dabei ein Lesefehler auf, so rotiert Stepper 1 die Diskette erneut neunmal und liest aufs Neue. Ist der Versuch, diesen Sektor zu lesen oder zu schreiben, fünfmal schiefgegangen, so wird eine Fehlermeldung ausgegeben, der Mo-



tor gestoppt und die letzte Position im RAM des Rechners zwischengespeichert.
Bei der nächsten Order zum Positionieren des Kopfes kann sich der Rechner das Zurückfahren auf Spur 0, Sektor 0 ersparen und arbeitet da weiter, wo er aufgehört hat.
Auf diese Positionierungsvorgänge hat der Anwender keinen Einfluß. Der Controller betätigt sich hierbei als Pfadfinder und weiß von vornherein, wo er Daten zu suchen hat.
Ihre Diskette hat nämlich auch noch ein Inhaltsverzeichnis (DIRECTORY/CATALOG), in dem nicht nur steht, welche Programme auf der Disk enthalten sind, sondern auch noch, für den Anwender nicht sichtbar, wo die Daten genau zu finden sind.
Die Position des Directorys auf der Diskette ist abhängig vom Format.

Wie Sie sehen, hat die kleine Diskettenstation ganz schön viel zu tun. Noch komplizierter verhält es sich beim Vortex-Format. Wenn der Schreib-/Lesekopf Spur 80, Sektor 9 erreicht hat, so muß er auf die Diskettenunterseite umschalten und dort weitersuchen. Aber die Grundlagen über die Hardware sind hiermit abgeschlossen. In Abb. 2 sehen Sie noch einmal den Aufbau einer formatierten Diskette und in Abb. 3 alle Bewegungen, die die Steppermotoren ausführen müssen.

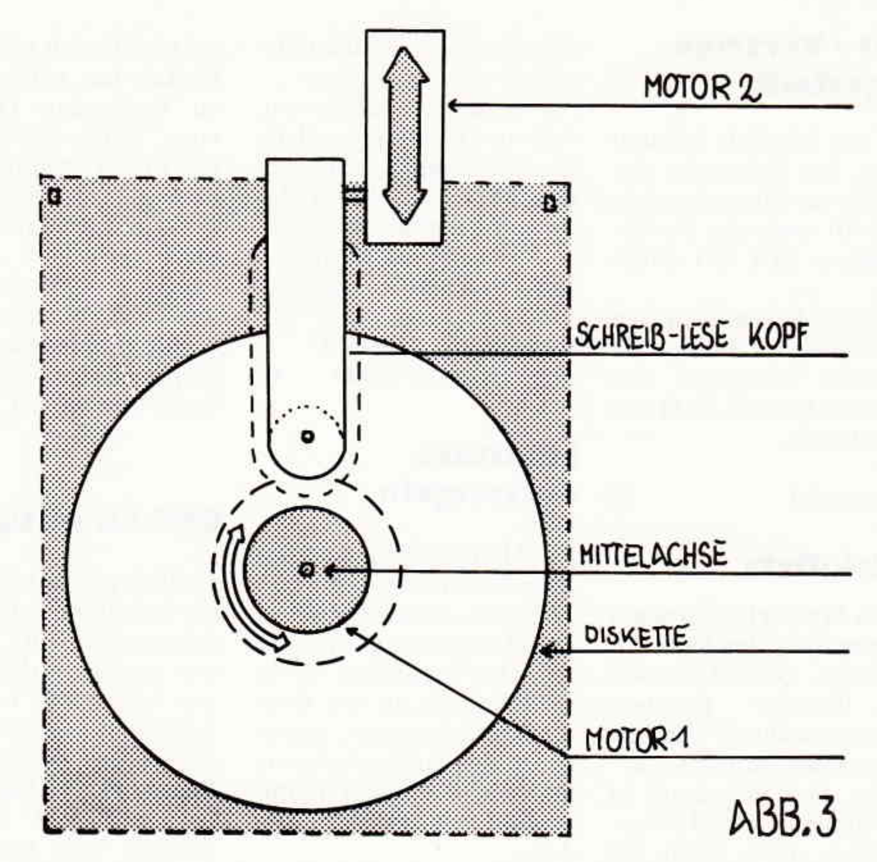


ABB.3

Im nächsten Teil geht es dann schon um die Programmierung des DOS und die Befehle, die es dem Anwender auf der Ebene einer Hochsprache wie Basic zur Verfügung stellt. (TM)

NLQ - Vertrieb eingestellt

Wie erst kürzlich bekannt wurde, hat Schneider den Vertrieb des Matrixdruckers NLQ 401 sowie des Formulartraktors SFT 401 eingestellt.
Das Nachfolgemodell, der DMP 2000, ist bereits seit längerem erfolgreich etabliert und löst den NLQ nun endgültig ab.
Info:
Fachhandel □

Quick-Data

Die Fa. Werder hat ein neues Programm für den Joyce angekündigt. Quick-Data soll die Bereiche Kundenstammverwaltung, Artikelstamm sowie Datenbank abdecken. Das Programm ist vollständig in CP/M 3.0 geschrieben (evtl. auch für 6128?) und steuert die GSX-Schnittstelle zur Erzeugung von Bildschirmgrafiken an. Des weiteren gibt es zahlreiche Druckroutinen für Nachnahmebelege, Postkarten, Etiketten etc.
Info:
Fa. Werder
2000 Hamburg 71 □

Programmierbares RS-232-Kabel

Lindy bietet jetzt ein programmierbares, fünfadriges Kabel an, mit dem viele RS-232/V.24-Verbindungen zu lösen sind. In jedem Stecker sind zehn kleine Schiebeschalter eingebaut; insgesamt lassen sich mehr als 1 Mio. Schaltvarianten realisieren. Das 2 m lange Kabel trägt beidseitig die genormten, 25-poligen Sub-D-Stecker und eignet sich für beinahe alle RS-232/V.24-Verbindungen.
Vertrieb: Fachhandel □

Festplatte für Joyce

In England wurde jetzt die erste Festplatte für den Joyce PC vorgestellt. Die sogenannte Amstore hat eine Speicherkapazität von 20 Megabyte; der Preis stand allerdings bei Redaktionsschluß noch nicht fest.
Auf Anfrage erfuhren wir, daß zur Zeit erst einmal die Marktchancen sorgfältig abgewägt werden, bevor das Produkt in größeren Stückzahlen hergestellt wird.
Info: Northern Computers Ltd.
Cheshire, WA6 6 RD
Tel:(GB) 0928/35700 □

Monitore entspiegeln

Als Alternative zu diversen Bildschirmfiltern gibt es nun ein Spray, das den Monitor eines Computers entspiegeln soll. Das "anti-reflect"-Spray wird einfach auf den Bildschirm aufgetragen; dieser kann nach der Beschichtung mit allen üblichen Reinigungsmitteln behandelt werden.
Der Preis für eine Dose "anti-reflect"-Spray liegt bei ca. DM 29,90.
Info: resco electronic
8900 Augsburg □

Prolog für CP/M 2.2

Das Thema "Künstliche Intelligenz" erfreut sich regen Interesses. Die Programmierung dieser komplexen Thematik wird vor allem in den Sprachen LISP und Prolog vorgenommen. Nun gibt es die "micro-Prolog"-Version für CP/M 2.2-Rechner, somit können auch CPC 464- und 664-Besitzer in den Genuß dieser recht unbekannten Programmiersprache kommen.
Info: Fa. Brainware
6100 Wiesbaden □

The Music-System

Rainbird, die Tochterfirma von Firebird, hat das Folgeprodukt ihre Debüt-Hits "The Music-System" angekündigt. Das Disketten-orientierte "Advanced Music-System" (AMS) soll zwei weitere Module enthalten: Eine Option zum Partitur-Ausdruck von Noten und Texten sowie einen Linker, um editierte Files zu verbinden. Die User vom "Music-System" können ihr Paket aufwerten, in dem sie es ohne Verpackung mit 10 Pfund zurücksenden.
Info:
Rainbird Software
Wellington House
Upper St.Martin's
Lane, London WC2H 9DL □

GSX-Literatur

Endlich gibt es Literatur zu der geheimnisvollen GSX-Schnittstelle von CP/M 3.0., was vor allem die Anwender von CPC 6128 und Joyce freuen dürfte.
Die englischsprachige GSX-Literatur von Digital Research besteht aus dem GSX-80 User Guide (Benutzerhandbuch) und dem GSX-Programmer's Guide (Programmierhandbuch).
Das Paket hat einen Preis von ca. DM 148,- und ist bei allen Schneider-Vertragshändlern erhältlich.
Info:
Schneider Data
8050 Freising □

Joyce-Schulung

Die Möglichkeiten von LocoScript auszunutzen, ist Ziel der Joyce-Schulung von IC. im Berliner Stadtteil Neukölln. Die individuell gestalteten Kurse finden in der Zeit von 19 - 22 Uhr statt, die Kosten belaufen sich auf ca. DM 228,-.
Die Veranstalter machen darauf aufmerksam, daß der Kurs solange dauert, bis das Lernziel erreicht ist.
Info:
Fa. Ihre Computerei
1000 Berlin 44 □

Elektronik- und Computertage Saar

Vom 5. - 7. September finden in der Kongreßhalle Saarbrücken die 3. Elektronik- und Computertage Saar als Verkaufs- und Informationsmesse statt.
Nachdem im vergangenen Jahr fast 120 ausstellende Firmen teilnahmen und fast 9000 Besucher gezählt wurden, kann dieses Jahr das Ergebnis noch verbessert werden.

Neben den kommerziellen Rechnern findet der Interessent auch eine große Auswahl an Zubehör und Fachliteratur.
Begleitet wird diese Ausstellung durch zahlreiche Vorträge und Demonstrationen.
Info:
Elektronik- und Computertage Saar
6620 Völklingen □

10-Fingersystem

Jetzt gibt es ein Lehrbuch und eine Programmdiskette zur sicheren Beherrschung der Tastatur im Zehnfingersystem. Es werden Texte vorgegeben, korrigiert und ausgewertet.
Das begleitende Lehrbuch zeigt anhand vieler Beispiele und Übungsaufgaben die Technik des Maschinenschreibens. Das Paket ist für CPC 664 und 6128 erhältlich, der Preis liegt bei ca. DM 69,-.
Info:
Verlag Mende
7514 Eggenstein □

Mirage Imager

Der Mirage Imager für alle CPC's ist ein Hardware-Zusatz zum Übertragen von Programmen von Kassette auf Diskette und umgekehrt. Auf einen Knopfdruck hin, wird das laufende Programm eingefroren, anschließend erscheint das Transfer-Menue. Der Mirage Imager wird voraussichtlich etwa DM 249,- kosten.
Info:
Mirage Microcomputers
5400 Koblenz 16 □



MIRRORSOFT Präsentiert
AUS DEM OSTEN
SAI COMBAT
Sai ist eine japanische Kampfart, bei der die Gegner versuchen, sich mit Stockschlägen in ihren Aktionen zu blockieren. Sie können mit Ihrer stockschwingenden, stockstechenden und stockschlagenden Figur 16 verschiedene Kampfbewegungen ausführen!
"Sai Combat ist ohne Zweifel das beste Kampfspiel, das ich je auf dem Spectrum gesehen habe ... Ein Spitzen-Produkt aus dem Hause Mirrorsoft". – Aktueller Software Markt
Schneider CPC (Gebrauchsanweisung in Deutsch). Spectrum 48K
und AUS DEM WESTEN
Spitfire 40
Sie beginnen als Flugschüler auf der Spitfire und müssen Ihre Fähigkeiten beweisen, bevor Sie sich bewähren dürfen. Sie sammeln Erfahrungen und speichern Ihre Flug-Logbücher auf Band oder Disk, um den begehrten Rang eines Fliegeroffiziers zu erreichen.
Das Spiel hat drei Stufen:
Übungsflüge – überwachte Starts, Flüge und Landungen, die Sie mit den vielen komplizierten Instrumenten vertraut machen. Freie Flüge – Sie steigern Ihre Manövrierfähigkeiten. Einsatzflüge – jetzt wird es ernst.
Schneider CPC. Spectrum 48K. Commodore 64 in Kürze für MSX. Atari.
RUSH WARE
Online with the trend.
Rushware
An der Gümpgesbrücke 24
4044 Kaarst 2
MIRRORSOFT
Purnell Book Centre, Paulton, Bristol BS18 5LQ



Die vierte Amstrad Computer Show, die vom 13. - 15. Juni 1986 im Londoner Stadtteil Hammersmith stattfand, brach alle bisherigen Rekorde. Besucher und Aussteller konnten drei tolle Computertage im Londoner (Hotel) Novotel erleben, die allerhand an Neuheiten zu bieten hatten.

Vierte Amstrad Show zeigt neue Wege

Als am 13. Juni die Pforten für die Amstrad Show geöffnet wurden, drängten schon zahlreiche Besucher in die Ausstellungsräume. Mit 70 Ausstellern war dies die bisher größte Amstrad Show in England. Gemeldet hatten sich allerdings weit mehr als hundert Aussteller, die aber aus Platzgründen nicht mehr untergebracht werden konnten, wie uns Derek Meakin von Database Publication versicherte.
Database Publication, Veranstalter zahlreicher Computermessen, ist auch Herausgeber der englischen Zeitschriften "Computing with the Amstrad" und neuerdings auch von "Amstrad User".
Neuer Rekord war auch die Ausstellungsfläche von etwa 3000 qm, die sich im Gegensatz zur vergangenen Amstrad Show mehr als verdoppelt hat, so Derek Meakin weiter.
Und wie zu erfahren war, soll die nächste Show im Oktober noch mehr Fläche und dann auch vier Ausstellungstage bieten.
England schwelt zur Zeit im absoluten Amstrad-Fieber; kein anderer Computer hat zur Zeit einen so gewaltigen Markt an Hard- und Software aufzuweisen.
So waren allein 20 neue Firmen in London angetreten, um ihre Produkte erstmals der breiten Öffentlichkeit zu präsentieren.

Joyce der Renner

Überraschend war festzustellen, daß der PCW 8256 bzw. 8512 (hierzulande als Joyce bekannt) der Engländer liebster Computer zu werden scheint. So werden allein in England zur Zeit jeden Monat 20000 Joyce verkauft; der Trend ist weiter steigend.
Vergleicht man nun einmal diese Zahlen mit den Verkaufszahlen hierzulande, lt. Schneider sind ca. 30000 Joyce verkauft, so können die Händler nur mit verschämten Blick auf die Insel hinüberblicken.
Versucht man die Gründe für diese doch sehr unterschiedlichen Verkaufszahlen zu erforschen, so fällt sofort das Verkaufsargument ins Auge.
Während der Joyce bei uns als kompaktes und preiswertes Textsystem angepriesen wird, gilt der PCW in England als preisgünstiger PC mit CP/M-Betriebssystem, mit dem man selbstverständlich auch eine gute Textverarbeitung erhält.

Diesem Umstand ist es auch zu verdanken, daß momentan eine wahre Softwareflut für den Joyce/PCW im Anrollen ist, während gute Anwendersoftware hierzulande nur schwer auszumachen ist.
Nun gut, könnte man meinen, was jetzt in England an Hard- und Software angeboten wird, läßt bei uns auch nicht lange auf sich warten. Leider ist dem nicht so!
Während die Produkte für die CPC-Computer fast ausnahmslos und ohne große Zeitverzögerung bei uns in den Handel kommen - die landläufige Meinung, daß die Engländer uns ein halbes Jahr voraus sind, stimmt schon lange nicht mehr - sieht die Situation bei Joyce-Produkten doch etwas anders aus.
Die vorgestellte Software - einige sehr gute Datenbanken, Finanzprogramme usw. - müssen zunächst komplett eingedeutscht werden, um bei uns überhaupt eine Anwendung zu finden.
So wurde z.B. ein LocoScript-Syntax-Checker vorgestellt, der die Orthographie überprüft und vorhandene Fehler anzeigt - aber wie gesagt, alles in englischer Sprache.
Um ein solches Programm ins Deutsche zu übertragen, gehen Wochen, wenn nicht gar Monate ins Land, und während dieser Zeit herrscht eben Ebbe. Es sei denn, die deutschen Softwarehersteller nehmen sich eigener Ideen an und stellen ähnliches auf die Beine.

Trend zur Anwendung

England gilt ja allseits nicht nur als Mutterland des Fußballs, sondern seit der Einführung des Sinclair ZX 80/81 - der erstmals für Otto-Normalverbraucher erschwinglich war - auch als Mutterland des Heimcomputers.
Der englischen Mentalität zufolge wurden die Heimcomputer über Jahre hinweg vorrangig zum Spielen verwendet; die führenden Softwarehersteller des Genres Computerspiel (z.B. Ultimate, Bubble Bus, Mikrogen etc.) kommen fast ausnahmslos aus England.
Völlig entgegengesetzt verlief dagegen die Entwicklung in der BRD; von Beginn an waren Anwenderprogramme gefragt, das Spiel galt und gilt allenfalls als gelungene Abwechslung.
Umso überraschender konnten wir beim Besuch der Amstrad Show feststellen, daß in England gerade eine Umbruchphase beginnt und sich dieser Markt zunehmend der sinnvollen Heim- und Business-Anwendung annimmt.
Dies mag mit Sicherheit auch daran liegen, daß die Amstrad Computer schon von der Konzeption viele Einsatzbereiche abdecken können



Günstige Zweitlaufwerke für die CPC-Computer gingen weg wie warme Semmeln.

und im Zusammenspiel mit entsprechender Soft- und Hardware zu echten Allroundtalenten herangereift sind.
In England ist alles anders – diese Aussage mag man aufgrund der neuen Begebenheiten nicht mehr so leichtgläubig hinnehmen.
Auf den zweiten Blick könnte man folgende Feststellung treffen: In England ist vieles anders!
Schon beim lockeren Rundgang durch die Ausstellungsräume fiel zwangsläufig das breit gefächerte Publikum auf, und – man höre und staune – das Thema Frauen und Computer scheint in England überhaupt kein Thema zu sein!
Wie selbstverständlich begegnet man Vertretern des weiblichen Geschlechts nicht nur als Besucher, sondern sieht sich auch Frauen als Standpersonal sehr oft gegenüber.
Dabei ist allerdings zu bemerken, das diese – oftmals sehr hübschen jungen Damen – nicht nur Poster und Sticker verteilen, wie das bei uns von Messen in Hannover oder München hinlänglich bekannt sein dürfte.
Vielmehr begegnete uns in London das weibliche Standpersonal als kompetenter Gesprächspartner in allen Fragen des Computergeschäfts; eine sehr wertvolle Erfahrung für uns, die sich doch sonst eher einer Männerdomäne gegenübersehen!
Daraus folgt logischerweise auch die Tatsache, daß die englischen Frauen eine gänzlich andere Beziehung zu der Computertechnik besitzen, als dies bei uns der Fall ist. So waren unter den vielen tausend Besuchern zahlreiche, weibliche Computerfans mit entsprechendem Fachwissen vertreten.

Neue Produkte

Wie bereits erwähnt, konnte an vielen Ständen neue Anwendersoftware für den Business-Bereich begutachtet werden, die sich hauptsächlich in Form von Datenbanken, Finanzprogrammen etc. zusammensetzte. Diese Programme hier im einzelnen zu erwähnen, scheint nicht sinnvoll, da zur Zeit keine Distribution in der BRD vorgesehen ist und englischsprachige Anwendersoftware sowieso nur schwer an den Mann zu bringen ist.
Eine Ausnahme ist lediglich das schon vom Sinclair Spectrum bekannte, universell einsetzbare Dateisystem Masterfile III. Wie wir erfahren konnten, ist die deutsche Version schon in Arbeit und wird demnächst auch hierzulande vertrieben.
Die bisher so erfolgreich vertriebene AMX-Maus von Advanced Memory Systems hat Zuwachs in Form von Software bekommen.
Zunächst werden die Programme Pagemaker und Magazine-Maker auf den Markt kommen, die an die legendären Print-Shop und News Room des C-64 angelehnt sind.
Damit wird es möglich, mit dem Eingabegerät Maus seine eigene Zeitung bzw. einzelne Zeitungsseiten zusammenzustellen.
In Kürze wird die AMX-Maus auch für den Joyce PC angeboten, der damit endlich eine komfortable Benutzeroberfläche erhält und um eine weitere, sinnvolle Anwendung ergänzt wird.
Neue Software gibt es inzwischen auch für das Grafiktablet Grafpad II, das von der Fa. Pizie Data vertrieben wird. In Kurzfassung einige Features der neuen Software:

- Druckertreiber zum Anpassen an gängige Drucker
- freie Rasterwahl
- Hardcopy in 2 Formaten mit Grauraster
- komplett in Maschinensprache, daher genauere Stiftabfrage u.v.m.

Besitzer des Grafpad II können die neue Software gegen Aufpreis bei Pizie Data ordern.
Das Grafiktablet wird ebenfalls für den Joyce angeboten, unterstützt jedoch die reine Vectorgrafik. Der voraussichtliche Verkaufspreis liegt bei ca. DM 550,–.
Ebenfalls an den Joyce angepaßt hat Electric Studio seinen EL-Lightpen; die Demos waren schon sehr beeindruckend. Dieser Lightpen wird schon bald unseren Handel erreichen, die deutsche Bedienungsanleitung steht kurz vor der Fertigstellung.
Biggles – so heißt einer der erfolgreichsten Filme, die zur Zeit in England's Kinos zu sehen sind. Kaum im Filmtheater – schon ist das Computerspiel zum Film fertiggestellt. Biggles ist ein mehrteiliges Arcade/Strategiespiel; eine ausführliche Review können Sie in unserer nächsten Ausgabe nachlesen.
Von Hisoft kommt Turbo Basic – ein Compiler, der Basicprogramme in Maschinencode übersetzt und die Ablaufgeschwindigkeit bis zu 80mal erhöht. Leider unterstützt Turbo Basic keine Fließkommaarithmetik und gilt daher in der Anwendung zumindest als eingeschränkt.
Dieser Basic Compiler wird voraussichtlich, wie alle anderen Hisoft-Produkte auch, von der Fa. Schneider vertrieben.
Ein weiteres, sehr interessantes Softwareprodukt entdeckten wir bei Timatic Systems, die bisher vor allem durch ihre Diskettenlaufwerke bekannt waren.
Das Programm Microdraft ist ein komfortables CAD-Programm, das unter CP/M 3.0 läuft und somit auch



Soft- und Hardware-News aus England. Der Markt um die Schneider/Amstrad Computer wächst ständig.



vom Joyce verarbeitet werden kann. Durch eine reichhaltige, selbst zu bestimmende Symbolbibliothek ist Microdraft in der Lage, auch komplizierte Schaltpläne zu erstellen und ganze Layouts zu erzeugen. Ein besonderer Clou dieses Programmes dürfte die Möglichkeit sein, von IBM-Computern erstellte CAD-Grafiken mit dem Schneider bzw. Joyce weiter zu verarbeiten. Natürlich hatte Timatic auch neue Hardware auf Lager. Vorgestellt wurde u.a. ein Eprom-Programmiergerät sowie ein Zweitlaufwerk für den Joyce.
Das im 5 1/4"-Format gehaltene Laufwerk hat eine Kapazität von 1MB und kostet ca. 210 engl. Pfd. (ca. DM 700,–). Dazu gibt es allerdings noch ein reichhaltiges Software-Paket, das u.a. Turbo-Pascal, dBaseII und Newword enthält und damit ein interessantes Angebot sein dürfte.



Zubehör für Joyce. Der Lightpen von Electric Studio erfreute sich großer Nachfrage.

Joyce-Spiele

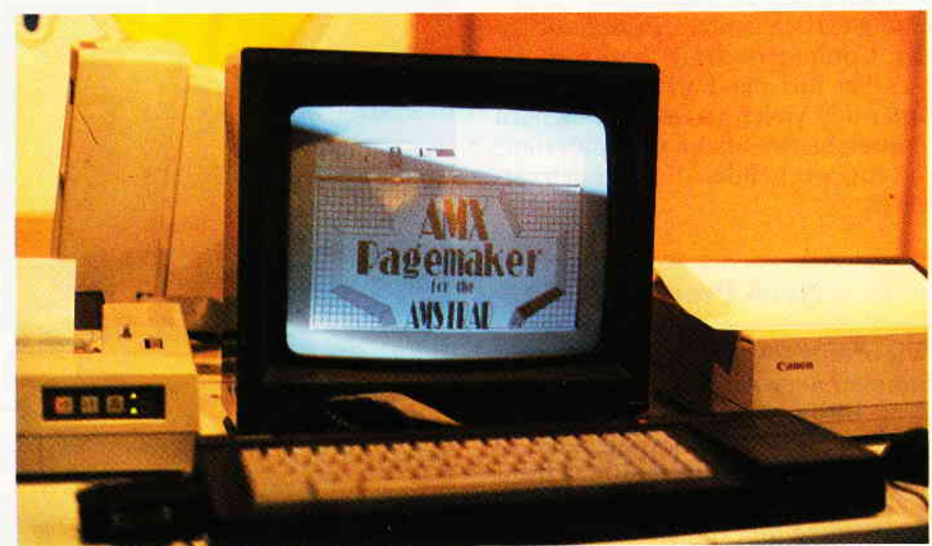
Daß auch der Joyce zur Abwechslung mal mit einem guten Computerspiel gefüttert werden möchte, scheint recht naheliegend. Und Englands Softwarehäuser arbeiten eifrig an Joyce-Spielen, die den Unterhaltungsbereich abdecken sollen. Das Schachprogramm Colossus 4, von CDS Software, wohl eines der spielstärksten überhaupt, ist bereits angepaßt. In einem kurzen Test konnten wir uns von der extremen Leistungsfähigkeit dieses Programmes überzeugen. Näheres dazu erfahren Sie in einer unserer nächsten Ausgaben.
Rainbird und Level 9 zeigten die ersten Grafikadventures für Joyce-Computer: Time and Magik, The Archers und Jewel of Darkness werden im Herbst auch bei uns erhältlich sein, die gezeigten Grafiken waren bereits sehr beeindruckend.

Vom Midi-Interface bis zur Festplatte

Die Musikfreunde unter den Schneider-Anwendern können sich ebenfalls über einige neue Produkte freuen. Die Fa. Vanguard Leisure stellte den Music Maestro vor, der für alle CPC's erhältlich ist. Das Paket beinhaltet einen Stereo-Verstärker, zwei 30W Lautsprecher und natürlich die entsprechende Software. Als Demo mußte Jean Michel Jarre's Synthesizer-Hit Oxygene herhalten, der in äußerst eindrucksvoller Art und Weise fast originalgetreu wiedergegeben wurde.

Endlich wurde auch einmal ein komplettes Midi-Interface gezeigt, mit dem mehrere Synthesizer vom Schneider-Computer angesteuert werden können. Electromusic Research, kurz EMR, hatte dann auch die erste Zusatz-Software parat, den sogenannten Miditrack Performer.
Wie uns versprochen wurde, sollen wir umgehend ein Testexemplar erhalten, das wir dann selbstverständlich ausführlich testen und vorstellen werden.
Erfahren konnten wir dann noch, das EMR schon bald eine Joyce-Version publizieren wird, auch hier laufen die Entwicklungen auf Hochtouren.
Sind Sie Discjockey, veranstalten oft Partys im Hobbykeller oder gefallen Ihnen die Lichteffekte Ihrer Stamm-Diskothek nicht, dann kann Ihnen geholfen werden.
Denn jetzt gibt es die computergesteuerte Lichtanlage (allerdings nur für CPC 6128) von Kelitronic. Mit diesem System können bis zu 96 Kanäle angesteuert werden; die Software ist entsprechend umfangreich.

Unseren Informationen zufolge gibt es zur Zeit drei Diskotheken in England, die mit diesem System in Verbindung mit CPC 6128 ihre gesamte Lichtanlage steuern.
Der Begriff Festplatte kursiert auch bei uns schon des längeren in der Branche herum. Northern Computers stellte nun eine 20MB Harddisc vor, die zum Preis von ca. 1300 Pfd. angeboten wird. Dieser Preis beinhaltet eine vollwertige Centronics-Schnittstelle, einen Printer Spooler sowie einen Druckerpuffer.
Spricht man über Hardwareprodukte zum CPC, kommt man an der Peripherie von dk'tronics nicht vorbei. Zum altbekannten Lightpen und Sprachsynthesizer sind nun zwei Speichererweiterungen (64k und 256k) und die sogenannte Silicon Disc hinzugekommen.
Wie wir erfahren konnten, sind die beiden Speichererweiterungen voll kompatibel zu der am Markt befindlichen Software, mit entsprechenden Erweiterungen anderer Hersteller hat man in England keine allzu guten Erfahrungen gemacht, wie uns von seitens Amsoft versichert wurde.
Die Silicon Disc von dk'tronics hat eine Speicherkapazität von 256 K und ist voll kompatibel zu den eben genannten Speichererweiterungen.
Diese Silicon Disk ähnelt vom Äußeren ebenfalls den Speichererweiterungen, bzw. der Box des Lightpens und Sprachsynthesizers und ersetzt quasi ein Zweit- oder Drittlaufwerk.
Voraussetzung für den Betrieb der Silicon Disc ist allerdings ein normales 3"-Disklaufwerk; die im ROM verankerte Software erkennt automatisch den entsprechenden Status (Laufwerk A;B;C). Die Silicon Disc ist vergleichbar mit einer RAM-Floppy; der



Die AMX-Maus bekommt Nachwuchs. Neben einem Utility-Pack kommen Page-Maker und Magazine-Maker.



Supersound mit dem CPC.
EMR zeigte ein Midi-Interface zum Anschluß an Synthesizer.

Speicherinhalt ist nach dem Ausschalten des Computers also gelöscht.
Einiges Aufsehen erregten zwei kleine, schwarze Boxen, die einfach in den Expansion Port der CPC-Computer eingesteckt werden. Die Firmen Romantic Robot (Multiface two und Mirage (Mirage imager) stellten ein Interface vor, mit dem auf Knopfdruck jedes beliebige Programm "eingefroren" und auf Kassette oder Diskette abgespeichert werden kann. Des weiteren kann eine genaue Programmanalyse vorgenommen werden; über zahlreiche Menues kann der jeweilige Status abgefragt und sogar geändert werden. Eine denkbar sinnvolle Anwendung wäre zum Beispiel das Ändern der Farben von Spielfiguren oder Hintergründen, was Besitzern von Grünmonitoren in diesem Fall zugute kommen würde.
Raubkopierer mögen hier eine echte Chance wittern, um Programme auf einfachste Art zu "knacken" und entsprechend weiterzugeben. Doch ganz so einfach ist das nun auch wieder nicht. Das sogenannte "Knacken" (z.B. Ändern des Copyrightvermerks) geht zwar recht schnell und einfach, hat aber zur Folge, daß die geänderten Routinen in das 8k umfassende RAM des Interface geschrieben werden. Um die modifizierte Software dann aber ablaufen zu lassen, wird wiederum das entsprechende Interface benötigt und schon ist die Grenze des Mißbrauchs erreicht.

Daten per Telefon

Betrachtet man einmal die vielfältigen Möglichkeiten der Kommunikation zwischen Computern, so fallen sofort die Begriffe DFÜ, BTX oder Datex-P.
Der englische Postservice Prestel, vergleichbar mit unserem BTX-Netz, hat wohl inzwischen so etwas wie eine Vormachtstellung erreicht. Heute schon nutzt fast jeder englische Betrieb, ja selbst die große Masse der Privatleute, die modernen Kommunikationstechniken in Verbindung mit einem Heim- oder Personalcomputer.
Die Vorteile liegen klar auf der Hand, die elektronische Kommunikation geht problemlos vonstatten und wird in England zu einem sehr günstigen Preis angeboten.
Schwierigkeiten mit der Erteilung von FTZ-Nummern kennt man in England nicht und überhaupt scheint die englische Post diese Entwicklung in keinster Weise zu blockieren.
Betrachtet man dagegen die Aktivitäten der deutschen Bundespost, kann man fast nur noch in Hohngelächter ausbrechen. Wieder einmal mehr ist der berühmt berüchtigte deutsche Bürokratismus Vorläufer einer Generation, die dem Weltstandard schon jetzt ein gutes Stück hinterherhinkt und Gefahr läuft, den so wichtigen Anschluß erneut zu verpassen.
Den bundesdeutschen Postlern würde ein Trip nach England sehr gut tun, um einmal zu sehen, wie ohne großen Aufwand eine Datenkommunikation der breiten Masse zur Verfügung gestellt werden kann.
Aber noch ist ja nicht aller Tage Abend und wer weiß, vielleicht kommen auch wir einmal in den Genuß dieser gebührengünstigen Kommunikation.

Fazit:
Die Organisatoren haben eine Super-Amstrad Show auf die Beine gestellt, die Publikum und Aussteller ausnahmslos begeisterte. Als Herausgeber der CPC International waren wir hocherfreut zu vernehmen, daß unsere Zeitschrift auch in England einen sehr guten Ruf hat und gern gelesen wird.
Unsere jüngsten Aktivitäten, Databox-Service, Hardware-Ecke und Goldene 7 finden bereits Nachahmer; es wurde weiterhin eine enge und freundschaftliche Zusammenarbeit mit allen Branchenvertretern vereinbart.
Übrigens wußten Sie, daß Ihre CPC International mit einer Gesamtauflage von 90000 pro Monat die auflagenstärkste Schneider-Zeitschrift der Welt ist?
Die größte englische Zeitschrift, Computing with the Amstrad, kommt dagegen nur auf etwa 70000 Auflage. Ein starkes Stück, oder wie die Engländer sagen würden: "The worlds biggest independend computer magazine for amstrad computers". (SR)



Die 20MB Festplatte von Northern Computers. Ein Vertrieb hierzulande ist noch nicht in Sicht.

Präzision im Selbstbau



Bild 1: Mit Geduld und handwerklichem Geschick...

CPC-Besitzer, die ihren Rechner zur Erstellung technischer Zeichnungen oder hochwertiger Grafiken verwenden, wollen ihre Werke natürlich auch in angemessener Qualität zu Papier bringen. Ein Matrixdrucker stößt dabei schnell an seine Grenzen. Die Alternative heißt hier: ein Plotter. Dieser jedoch stößt in Preisbereiche vor, die leicht das Doppelte des Anschaffungspreises für den Rechner ausmachen können. Seit einiger Zeit werden allerdings einige Flachbettplotter in ausreichender Qualität auch als Bausätze angeboten – und dies zu wahren Dumpingpreisen.

Einer dieser Plotter ist der PL 22/B der Firma Kopisch. Dieser DIN-A3-Flachbettplotter ist sowohl als Bausatz als auch fertig aufgebaut erhältlich. Beide Versionen werden mit einer fertig aufgebauten Prozessorkarte ausgeliefert, die die Zeichenmaschine mittels komfortabler Software zu einem wahrlich intelligenten Peripheriegerät machen.

Bild 1 bietet dem geneigten Selbstbauer einen ersten Eindruck, welche Erwartungen ein gelungener Selbstbau an seine handwerklichen Fähigkeiten stellt. Die benötigten Einzelteile waren vollzählig und sorgfältig verpackt. Dank einer recht ausführlichen in vier Baugruppen unterteilten Bauanleitung war der Zusammenbau kein Problem. Die Teile waren paßgenau und ließen keinen Zweifel an ihrer Zugehörigkeit. Allerdings sei an dieser Stelle darauf hingewiesen, daß ein Plotter ein Präzisionswerkzeug darstellt und der Zusammenbau doch einige Grundkenntnisse der Mechanik erfordert, um die guten Daten des Gerätes nicht durch mangelhafte Arbeit zu verschlechtern. Ein kritischer Punkt hierbei ist die korrekte Ausrichtung der Führungen für X- und Y-Achse, die mit Geduld und Sorgfalt durch Einstellung der Linearkugellager vorgenommen werden muß. Ein arbeitsreiches Wochenende sollte allerdings ein zufriedenstellendes Ergebnis zur Folge haben.

Bild 2 zeigt schließlich den Lohn der Mühe – ein durchaus ansprechendes Gerät, welches in den technischen Daten den Vergleich mit wesentlich teureren Produkten nicht zu scheuen braucht.

Ein möglicher Pferdefuß sei hier nicht verschwiegen: Der Anwender wird vorläufig die Treiberroutine für den PL 22 selbst schreiben müssen. Softwarehersteller sind natürlich geneigt, die in ihren CAD- oder Grafikprogrammen enthaltenen Plottertreiber an die Produkte großer Hersteller anzupassen, um dadurch auch größere Verbreitung ihrer Programme zu erreichen. Das Betriebssystem des PL 22 allerdings stellt eine große Fülle von komfortablen Steuer- und Kontrollbefehlen zur Verfügung, die allesamt durch 'PRINT # 8' angesprochen werden können, so daß auch von Basic aus ein universelles Treiberprogramm leicht herzustellen ist.

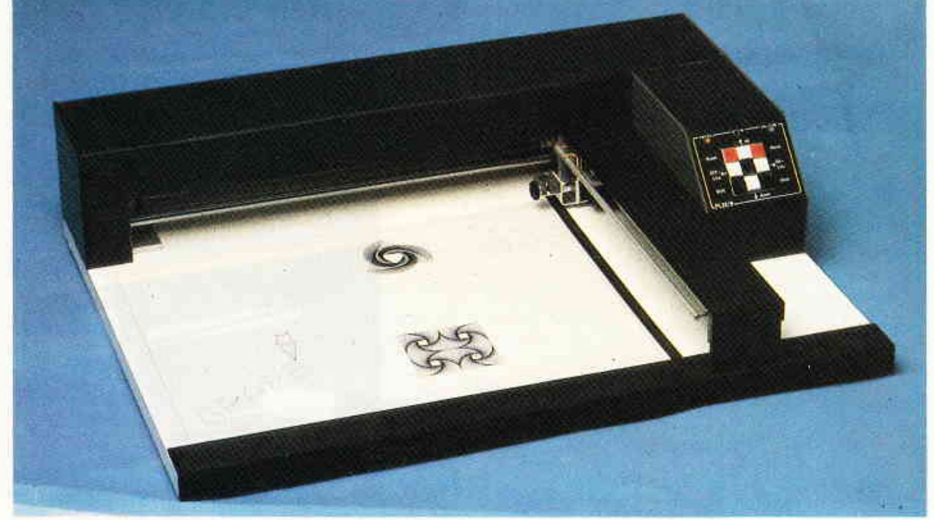


Bild 2: ... entsteht ein DIN A 3-Plotter im Selbstbau

Die technischen Daten:

Software:

Betriebssystem 'GRAPE firm' mit vielen Sonderroutinen wie:
- Einstellung der Zeichengeschwindigkeit
- Schrift kann in Größe, Neigung, Drehung beliebig verstellt werden
- mehrere Strichlierungsarten
- Interpolation von Kurven; der Digitaleffekt bei Kreisen entfällt
- beliebiger stufenloser Zoom
- automatische Erstellung von 3D-Grafiken
- Befehle werden per 'LPRINT' oder 'PRINT #8' übergeben

Hardware:

DIN A3-Plotter mit Z 80 Coprozessor
- Anschluß per Centronicsport
- Auflösung 0,0625 mm
- max. Zeichengeschwindigkeit 60mm/sec
- spielarme Linearkugellager

Preise:

Mechanikbausatz	ca. 620,– DM
Prozessorkarte	ca. 390,– DM
Netzteilbaus.	ca. 200,– DM
Gehäuse	ca. 100,– DM
GRAPE Software	ca. 300,– DM
Komplettgerät der höchsten Ausbaustufe:	ca. 1575,– DM

Info: Walter Kopisch
Plotter- und Grafiksysteme
7712 Blumberg

(ME)

WIR SIND IHR STARKER



PARTNER



TELEFONISCHE BESTELLUNG: 07191/1528/29

Hardware:

3" 1-Megabyte-Laufwerk für Joyce	698,–

Alle Vortex-Erweiterungen ab Lager lieferbar.
Vortex-Diskettenlaufwerke:
5 ¼" Einfachlaufwerk mit 708 KB formatiert, aufrüstbar zum Doppellaufwerk (548,– DM) mit resistenten Assembler, Disassembler, Monitor und RSX-Befehlen. Anlegen von Random-Dateien unter Basic.

F1-S 5 ¼" Floppy	1198,–
M1-S 3 ½" Floppy	998,–
F1-D 5 ¼" Doppellaufwerk	1698,–
M1-D 3 ½" Doppellaufwerk	1498,–
F1-X 5 ¼" Zweitlaufwerk 708KB, ROM-Software	758,–
M1-X wie F1-X, jedoch 3 ½"	758,–
Aufrüstkit für 80 Spur Doppellaufwerke als X-Laufwerk	298,–
Adapterkabel zum Anschluß der Schneider-Floppy an F1-S (M1-S) oder F1-D (M1-D) als 2. Laufwerk	58,–

Software:

Tuba Ruba	C 39,—
Worldcup Carneval	C/D 39,–/59,—
Soccer	C/D 39,–/59,—
Shape & Sound	D 59,—
Meltdown	C/D 29,–/49,—
Doors of Doom	D 59,—
Thing on a Spring	C/D 29,–/49,—
Glass	C 39,—
Classic Invaders	C/D 15,–/35,—
Gilligans Gold	C 33,—
Night Gunner	C/D 34,–/54,—
Speeky Harold	C 12,80
C-Basic Delta	C 12,80

Für Joyce PCW 8256/8512

Profirem professionelles Geschäftssoftware-Paket mit Lagerdatei, Adressdatei, Sofort-Speicherfakturierung.
Preis: **178,— DM**

Fibuking professionelle Finanzbuchhaltung
- 60 definierbare Konten
- Bilanzauswertung
- einfache Buchungseingabe
- Ausdruck von Kontenrahmen + Kontenblöcken.

Preis: **138,— DM**

Spezialprospekt anfordern.

Wichtiges Zubehör:

Abdeckhauben aus grauem Kunstleder	
Konsole 464/664/6128	19,80
Floppy DD1/FD1	16,80
Floppy Vortex FD-1	19,80
Monitor grün	26,80
Monitor color	29,80
Drucker	19,80
Drucker (NLQ/DMP)	19,80
Druckerkabel 464/664	48,50
Druckerkabel 6128	58,50
Lightpen für CPC 6128	139,—
Farbband (black) für Okimate 20	14,—
Farbband (colour) für Okimate 20	17,—
Schneider Joystick	39,50
Disketten 3"	à 12,—
Disketten DS/DD 5 ¼"	10 39,—
Traktor für NLQ 401	79,50
Farbband für NLQ 401	14,80
Farbband für DMP 2000	19,80
Schutzhaube Rauchglas f. Konsole 464/664	35,—
Druckerständer Rauchglas	98,—
Diskbox 3"	39,80
RS 232 (V 24)	168,—
Telefonmodem	298,—

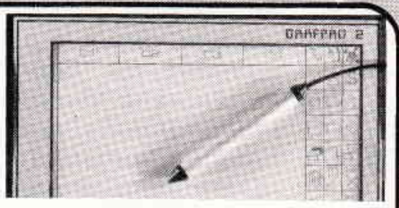
Software:

Stairway to Hell	C/D 29,–/49,—
Frankie goes to Hollywood	C 36,—
Ping Pong	C/D 35,–/49,—
Colossus Chess 4.0	C/D 39,–/59,—
Spindizzy	C/D 39,–/59,—
The Graphik Adventure Creator	C/D 79,–/89,—
Forbidden Planet	C/D 39,–/59,—
Eden Blues=Doomsday Blues	C/D 39,–/59,—
Crafton & Xunx=Get Dexter	C/D 39,–/59,—
Way of the Tiger	C/D 39,–/59,—
Souls of Darkon	C 39,—
Saboteur	C/D 39,–/59,—
Samantas Fox Strip Poker	C/D 39,–/59,—
Winter Games	C/D 39,–/59,—
Hexenküche II	D 49,—
Turbo Esprit	C/D 39,–/59,—
Fairlight	C/D 39,–/59,—
Tomahawk	D 49,—
Hacker	C/D 36,–/59,—
The Music System	C/D 69,–/79,—
Theatre Europe	C/D 39,–/59,—
Tau-Ceti	C/D 39,–/59,—
Battle of the Planets	39,—
Working Backwards	D 59,—
Rocky Horror Show	C/D 29,–/47,—
Marsport	C/D 39,–/49,—
Who dares wins II	C/D 36,–/59,—
Grand Prix Rallye II	D 59,—
Cluedo	39,—
Yie are Kung Fu	C/D 39,–/59,—
Fryday the 13th	C/D 39,–/69,—
Zorro	39,—
Gyroscope	34,—
Lords of Midnight	C/D 49,–/59,—
Lord of the Rings	C 59,—
Sweevos World	C/D 29,–/59,—
The Scout steps out	C/D 39,–/59,—
Death Wake	34,—
Tornado low level	C/D 39,–/59,—
Fighting Warrior/Exploding Fist	D 69,—
Bombjack	C 39,—
The Devil's Crown	C/D 39,–/59,—
3D Cyrus (Schach)	C/D 39,–/49,—
3D Grand Prix	C/D 39,–/49,—
Super Pipeline	C/D 36,–/56,—
Codename Mat II	C/D 29,–/59,—
Sorcery	29,—
Sorcery+ (Disk 3")	59,—
Frank Bruno's Boxing	C/D 29,–/49,—
Spy vs Spy	C/D 39,–/49,—
Elite	D 79,—
Spitfire 40	C/D 39,–/49,—

Graffpad II

- Auflösung: 1280 x 1024 Bildpunkte
- Abweichung: ± 1 Pixel
- Zeichenfläche: DIN A4
- hochauflösende Graphik
- Schaltungsdiagramme
- CAD
- Abmessungen: 350 x 260 x 12 mm
- Optionswahl
- Formeingabe
- Bilderspeicherung und Abruf
- Preis für 464, 664 **278,–**
- Preis für 6128 **298,–**
- Anschlußadapter f. 464-3" u. 6128 **48,–**
- mit deutschem Handbuch

Fordern Sie kostenlosen Prospekt an!
Sie werden begeistert sein!



AMX-Mouse

Mit Software (Zeichenprogramm + Basicerweiterung) Ähnlich wie bei Macintosh und Atari 520 ST für CPC 464, 664 und 6128.
Komplett
mit deutschem Handbuch **DM 278,—**

Software:

Barry Mc Guigan's Boxing	C/D 39,–/69,—
Mindshadow	C/D 39,–/59,—
Cheops	36,—
Spitfire 40	C/D 39,–/59,—
Hyper Sports	C/D 37,–/69,—
Night Shade	35,—
Heavy on the Magic	39,—
Blade Runner	35,—
Monty on the Run	C/D 39,–/59,—
Vokabeltrainer	C/D 49,–/59,—
Verbentrainer	C/D 39,–/49,—
Rock'n Wrestle	39,—
Red Hawk	35,—
Blue Ribbon	D 59,—
They sold a Million II	C/D 39,–/59,—
Movie	C/D 35,–/49,—
Ballblazer	C/D 39,–/59,—
Back to the Future	C/D 39,–/59,—
Rescue on Fractalus	C/D 39,–/59,—
Batman	35,—
Bounder	35,—
Fibuking (Finanzbuchhaltung)	136,—
Fakturem (Speicherfakturierung)	98,—
Profirem (Faktu, Lager, Adressen)	178,—

STAR-WRITER I (464 / 664 / 6128)

- Textverarbeitung
- Grafikprogramm
- DIN-Tastatur
- Serienbriefe
- Blockoperationen
- Trennvorschläge
- Suchen und Ersetzen
- Adressverwaltung
- DFÜ-Programm
- PULL-DOWN Menues
- Fließtexteingabe
- Block- und Flattersatz
- Bausteinverarbeitung
- Kopf- und Fußzeilen
- mit ausführlichem Handbuch nur ***198,-***

Händleranfragen erwünscht
Schulen erhalten Sonderkonditionen

Potsdamer Ring 10 · 7150 Backnang ☎ 07191/1528-29



COMPUTER-ELEKTRONIK

Der Diamant von Rabenfels
Reise durch die Zeit
Sherlock Holmes
Auftrag in der Bronx
Drachenland
Die Insel der Smaragde
Das Pharaonengrab
je Adventure
C/D DM 39,--/49,--

Software:

Kung-Fu-Master	C/D 39,–/59,—
The fith Axis	C/D 39,–/59,—
2112 AD	C/D 39,–/59,—
Shadowfire	C 35,—
V – this is it	C/D 39,–/59,—
N.O.M.A.D.	C 35,—
Zoids	C/D 35,–/59,—
Project Future	C 39,—
Biggles	C 39,—
Equenox	C/D 39,–/59,—
CORE	C 39,—
Arnheim	C 39,—

Software für Joyce:

LocoScript (Kurzanleitung m. Übungsdisk)	29,50
Prompt (Dateiverwaltung m. Listengenerator)	69,—
DR-DRAW	199,50
DR-GRAPH	199,50
Business-Pack (Adress-Lager- und Fakturmanager)	199,90
Computer-Wörterbuch (Englisch)	149,90
3-D Clock Chess (Schachprogramm)	79,90
MICA	198,—
RH-DAT (relative Dateiverwaltung)	89,—
Turbo-Adress (1400 Adressen)	169,—
Vokabeltrainer	69,—
Verbentrainer	59,—

Joyce:

Bildschirmfilter	89,—
3"-Diskette CF2-DD	19,80
Parallel- u. Seriellschnittstelle	148,—
Farbband	24,80

Für Joyce PCW 8256:

Megabytelaufwerk	698,—
Aufrüstkit von 256KB auf 512KB	149,—
DATEI-STAR	98,—
STAR-MAIL	98,—
FIBU-STAR+	298,—
STAR-BASE	198,—

O Hiermit bestelle ich per Nachnahme:

Vorname, Name

Straße, Hausnummer

PLZ, Ort

Telefon-Nummer

Datum, Unterschrift

EPROMS am CPC - die Software

Wie versprochen, erscheint diesmal das schlüsselfertige Programm für ein EPROM.
Es stellt 24 RSX-Befehle zur Verfügung und die Möglichkeit, automatisch Änderungen an den Voreinstellungen des Betriebssystems vorzunehmen.

Wie man das Programm ins EPROM kriegt

Starten Sie das BASIC-Programm "Patch" (Listing 1). Dieses Programm fordert Sie zur Beantwortung einiger Fragen auf, und setzt ihre Antworten in sogenannte "Patches" um, d.h. es poked gemäß Ihren Wünschen das im Speicher befindliche ROMSOFT-File.
Am Schluß generiert "Patch" ein neues ROMSOFT-File, das für die Verwendung im EPROM geeignet ist.
Diejenigen, die mit allen Voreinstellungen des ursprünglichen ROMSOFT-Files einverstanden sind, können sich das Laufenlassen von "Patch" auch sparen.
Schließen Sie jetzt den EPROMMER aus dem zweiten Teil dieses Berichtes an und legen Sie ein EPROM in die Programmierfassung. Sie sollten jetzt auch die Programmierspannung von 21V anlegen, und dann ohne Pause "ROMOUT" (Listing 2) starten. "ROMOUT" wird jetzt einige Minuten laufen und dabei die Adressen anzeigen. Wenn diese Prozedur fertig ist, schalten Sie zuerst die Programmierspannung ab und dann den Rechner. Jetzt können Sie das EPROM in die Schaltung aus Teil 1 des Berichtes einsetzen und anschließen.
Assemblerprogrammierer können das ROMSOFT-File vor dem Einbrennen natürlich beliebig ändern oder erweitern; sicher eine besonders reizvolle Möglichkeit, selbstgestrickte RSX-Kommandos zu verewigen.
Beachten Sie den Unterschied zwischen den zwei ROMSOFT-Files vor und nach dem Patchen; "ROMOUT" braucht das Gepatchte und "Patch" das Original.
Denjenigen, die nicht über einen Assembler verfügen, möchte ich zum Kauf der Databox raten. Das auf ihr erhältliche ROMSOFT-File enthält als kleines Extra eine besonders ausführliche Anleitung zu den Befehlen im ROM (RSX-Info).
Als problematisch erwies sich das Zusammenspiel von EPROM und Disklaufwerk. Möglicherweise ist der ungepufferte Bus des CPC nicht mehr stark genug, um beides korrekt anzusteuern.
Mein CPC verfiel auf jeden Fall in ein größeres Chaos, als ich beide Bausteine gleichzeitig anschloß. Sollte einer der Leser Abhilfe wissen, so wird sein Rat dankbar angenommen. Bitte beachten Sie in diesem Zusammenhang mögliche Korrekturen in den kommenden Heften.

Die Befehle

Zunächst die im "Patch"-Programm veränderbaren Einstellungen und ihre Defaults (Voreinstellungen):
- Mode, Border, Ink 0 (Default: Mode 2, Border 0, Ink 0,0)
- Die Einschaltmeldung (keine Vorgabe)
- Programmtextkomprimierung an/aus (Default: an)

Sollten Sie Programmtextkomprimierung "an" gewählt haben, so werden überflüssige Spaces in Ihrem Programmtext eliminiert.
Fest eingestellt ist die Belegung der Funktionstasten im Zehnerblock <0> <.> <enter> <1> <2> und <3> auf <Shift>-Ebene:
<0> : LIST
<.> : EDIT
<enter> : RUN cr
<1> : CAT cr
<2> : AUTO cr
<3> : CLS cr
cr steht für Carriage Return, gleichbedeutend mit ENTER.
Diskettenbesitzer werden weiterhin gefragt, ob Sie CP/M starten wollen. Vergessen Sie nicht, daß Sie eine CP/M-Diskette einlegen müssen, bevor Sie diese Wahl mit <ENTER> annehmen.
Lehnen Sie CP/M mit einer beliebigen anderen Taste ab, so werden Sie gefragt, ob Sie "Turbodos" aktivieren wollen. Wenn Sie mit <ENTER> annehmen, wird das AMSDOS so gepatcht, daß die Diskoperationen ca. 20 % schneller sind als bisher.
Besonders wenn Sie Laufwerke von Fremdherstellern verwenden, sollten Sie aber ablehnen (jede andere Taste), wenn es nicht so sehr auf Geschwindigkeit ankommt. Turbodos wird in CPC International 1/86 näher erläutert.

Speicherplatz

HIMEM wird durch ROMSOFT nur um 4 Byte verringert (diese 4 Byte benötigt das Betriebssystem zur Verwaltung der RSX). Es sollten also annähernd alle Programme, wie bisher funktionieren.
Es stehen 24 neue Befehle zur Verfügung, die im folgenden einzeln erläutert werden.

INFO:
Gibt einige Informationen zu den neuen Befehlen aus. Der Text entspricht bei der ROMSOFT-Version der Databox etwa dem, den Sie gerade lesen.

TB:
Zeigt die Tastaturbelegung der Funktionstasten an.

NON:
NLQ ON;stellt am Drucker NLQ ein, wenn dieser angeschlossen und aktiv ist. Andernfalls wird eine Fehlermeldung ausgegeben.

NOFF:
Stellt NLQ wieder ab.

HC:
Gibt eine Texthardcopy auf den Drucker aus, wenn dieser angeschlossen und aktiviert ist.
Die Hardcopy kann mit <ESC> abgebrochen werden (etwas länger drücken).
Alle bis jetzt genannten Befehle arbeiten ohne Parameter, d.h. alleine durch das Eintippen des Namens mit einem <SHIFT> <Klammeraffe> davor und anschließendem <ENTER>.

Befehle mit Parametern

HOCH,byte:
Scrollt den Bildschirm eine Textzeile hoch und fällt den neuen Bereich mit "byte". Bei richtiger Codierung entspricht "byte" einer Farbe, andernfalls ergibt sich ein Muster. Bitte experimentieren und in CPC International 11/85, S.34 nachlesen.

RUNTER,byte:
Scrollt den Bildschirm eine Textzeile runter.

NEXTLINE, Adresse Integervariable:
Dieser Befehl berechnet die Bildschirmadresse, die um eine Linie unter der angegebenen liegt. Die ursprüngliche Adresse muß in einer Integervariablen stehen, deren Adres-

se mittels Klammeräffchen an "Nextline" zu übergeben ist.
"Nextline" schreibt die neue Adresse in die Variable zurück.
Beispiel: 10 intvar%=&C000
20 |nextline,@intvar%

PRELINE, Adresse Integervariable:
Wie "Nextline", nur, daß eine Linie hochgerechnet wird.

NEXTBYTE/PREBYTE, Adresse Integervariable:
Berechnet das Byte, das rechts/links neben dem angegebenen auf dem Bildschirm steht. Bedienung wie "Nextline".

RDCHAR, Adresse Integervariable:
Liest das an der Cursorposition auf dem Bildschirm stehende Zeichen und übergibt den ASCII-Wert des Zeichens in der Integervariablen.
Beispiel:
10 intvar%=0:REM muß einmal benutzt worden sein
20 locate 20,20
30 |rdchar,@intvar%
40 locate 20,21:?chr$(intvar%)
Das Beispiel kopiert das Zeichen auf 20,20 nach 20,21.

FILL, byte, Höhe, Breite, Abstand von links, Abstand von oben:
Dieser Befehl fällt einen Bildschirmbereich mit der angegebenen Höhe und Breite (mindestens 1) und einem Abstand von links und von oben (mindestens 1) mit dem Byte. Wiederum ergibt sich bei richtiger Kodierung von "byte" eine Farbe. Hierzu siehe "HOCH".

DEL, byte:
Löscht den Bildschirm mit "byte" (vgl. HOCH). Dieses Löschen geht absichtlich langsam, weil's dann gut aussieht.

Kurze Erläuterung des Klammeräffchens:
Das "Äffchen" ist eine BASIC-Funktion und gibt die Adresse einer Variablen zurück. Bei Stringvariablen ist es die Adresse des Stringdescriptors.

SCRIPT, (Einrückung):
Durch "Script" geht der CPC in einen Schreibmaschinenmodus über, wenn ein Drucker angeschlossen ist. Um das "Papier" besser erkennen zu können, wird der Border etwas heller eingestellt. Der Cursor steht 4 Zeichen vom Rand entfernt, um die Einrückung zu symbolisieren, die bei der Druckerausgabe dem Text vorangestellt wird. Die Angabe der Einrückung ist optional; die Voreinstellung ist 5. Die Einrückung kann nicht größer als 15 gewählt werden. Der am Bildschirm editierte Text wird nach <Enter> ausgegeben. Es empfiehlt sich, <Enter> bereits vor dem Ende einer Bildschirmzeile zu drücken, da beim automatischen Zeilenumbruch des Druckers die Einrückung nicht berücksichtigt wird.
Das beste Bild ergibt sich, wenn man schon einige Zeichen vor dem Ende der Zeile <Enter> drückt (gewissermaßen eine Einrückung von rechts).

Die Cassettenbefehle RDCAS und WRCAS:
Sie erlauben ungeblocktes Ablegen und Laden von Binärfiles auf Cassette mit fast beliebiger Baudrate.
Interessant für lange Spiele mit sonst ewigen Ladezeiten. Derart gespeicherte Programme müssen mit "Call adresse" gewissermaßen von Hand gestartet werden.
R(ea)dcas dient zum Laden und Wr(ite)cas zum Ablegen.

WRCAS, Startadresse, Länge, Synchronzeichen, Rate:
Startadresse ist die Adresse, an der das Programm beginnt; Länge ist die absolute Länge des Programms. Das Synchronzeichen ersetzt den nicht angebbaren Namen; deshalb merken. Die Baudrate errechnet sich aus:
Baudrate = 333333 / rate
Z.B. ist Rate = 111 für 3000 baud.
Erfahrungsgemäß funktioniert dies höchstens bis etwa 3500 baud.
Bitte ausprobieren.

RDCAS, Startadresse, Synchronzeichen:
Das Synchronzeichen sollte das gleiche wie beim Ablegen sein.
Auf die Baudrate stellt sich Rdcas selbsttätig ein.

Die Discbefehle PUT und GET:
Diese Befehle ermöglichen direktes Beschreiben und Lesen einzelner Sektoren der Diskette. Damit läßt sich leicht ein Discmonitor oder eine relationale Datenbank programmieren. Im Umgang mit diesen Befehlen (besonders PUT) ist aber Vorsicht geboten, um keine Dateien zu zerstören.

Besonders wichtig:
Der Sektor ist mit einem Offset anzugeben, mit dessen Hilfe AMSDOS CP/M und Datendisketten unterscheidet.
Die Sektoren sind wie folgt numeriert:
CP/M: &41 bis &49
Data-only: &C1 bis &C9
Es empfiehlt sich, den Puffer oberhalb HIMEM anzulegen, um Zerstörung von Programmen zu vermeiden. Prinzipiell darf er aber überall liegen (so lassen sich tolle Overlays programmieren).

GET, Laufwerk, Track, Sector, Pufferadresse:
Dient zum Laden von Disc. Bei Laufwerk: 0 für A und 1 für B.
Tracks von 0 bis 39. Sektoren von 1 bis 9 PLUS Offset! Pufferadresse für 512 byte RAM (oberhalb HIMEM).

PUT, Laufwerk, Track, Sector, Pufferadresse:
Dient zum Ablegen auf Disc. Sonst alles gleich wie bei GET.

Die Monitorbefehle

Die folgenden Befehle bilden zusammen einen "Minimonitor":
DUMP, Start, (Ende), (Drucker)
Dump listet den RAM-Inhalt von Start bis Ende. Ende ist optional, die Voreinstellung ist 256 Byte (eine Seite). Das dritte Argument ist ein Scheinargument, darf also jeden Wert annehmen. Alleine durch sein Vorhandensein wird die Liste auf den Drucker ausgegeben, wenn es geht. Das Auflisten läßt sich mit <SHIFT> anhalten, solange es gedrückt bleibt.
MODIFY, Start
Dient zur Veränderung des RAM ab der Startadresse. X beendet Modify. <Enter> behält den alten Wert bei.
FLOOD, Start, Länge, byte
Flood fällt den Speicherbereich ab Start mit der angegebenen Länge mit "byte".
MEDIT, adresse
M(emory)edit dient zum Ablegen von Text (max. 255 Zeichen) im RAM. Der Text wird mit einem Nullbyte abgeschlossen.
SEARCH, Start, Länge, byte
Search sucht nach "byte" ab Start bis Start + Länge und gibt die jeweilige Adresse mit den folgenden 16 Bytes aus. Die Ausgabe kann mit <SHIFT> angehalten werden.
Die Monitorbefehle ergeben zusammen mit GET und PUT praktisch schon einen Discmonitor für interaktiven, d.h. nicht programmierbedürftigen Betrieb.
Das Ergebnis einer RAM-Modifikation (z.B. mit Modify) kann mit der B-Option des Savebefehls auf Cassette oder Disc gespeichert werden und mit 'Load ("Name", adresse)' an beliebiger Stelle wieder eingelesen werden (Adresse ist optional).

(J. Hüpper/ME)

```
10 MEMORY &3FFF
20 LOAD"romsoft.bin",&4000
30 FOR wertadresse=&4000 TO &7FFF
40 wert=PEEK(wertadresse):OUT &FBFB,wert:PRINT HEX
$(wertadresse),
50 FOR zeit=1 TO 50:NEXT zeit
60 NEXT wertadresse
70 END
```

Listing 1

```
10 MEMORY &3FFF
20 LOAD"romsoft",&4000
30 MODE 2
40 PRINT"********************* PATCH - PROGRAMM RO
MSOFT **************************"
50 PRINT:PRINT
60 INPUT "Welcher Mode soll eingeschaltet werden ?
 0,1,2 : ",m
70 IF m<0 OR m>2 THEN PRINT"Eingabefehler !":GOTO
60
80 POKE &40CE,m
90 INPUT "Welche Farbe soll der Border annehmen ?
0 - 27 : ",b
100 IF b<0 OR b>27 THEN PRINT"Eingabefehler !":GOT
O 90
110 POKE &40D3,b:POKE &40D4,b
120 INPUT "Soll die Programmtextkomprimierung akti
v sein ? j/n : ",k$
130 IF LOWER$(k$)="n" THEN POKE &40E9,0:GOTO 160
140 IF LOWER$(k$)="j" THEN GOTO 160
150 INPUT "Soll die Programmtextkomprimierung akti
v sein ? j/n : ",k$
160 IF LOWER$(k$)="n" THEN POKE &40E9,0:GOTO 190
170 IF LOWER$(k$)="j" THEN GOTO 190
180 PRINT"Eingabefehler !":GOTO 150
190 PRINT:PRINT"Geben Sie jetzt die Einschaltmeldu
ng ein :"
200 LINE INPUT "",em$
210 FOR lf=1 TO LEN(em$)
220 as=ASC(MID$(em$,lf,1)):POKE &42C9+lf,as
230 NEXT
240 SAVE"romsoft",b,&4000,&3800
250 END
```

Listing 2

```
10 DATA &01,&00,&00,&00,&51,&C0,&C3,&CA,&C0,&C3,&C
A,&C3,&C3,&D8,&C3,&C3, 2256
20 DATA &E6,&C3,&C3,&F7,&C3,&C3,&04,&C4,&C3,&11,&C
4,&C3,&7A,&C4,&C3,&B3, 2752
30 DATA &C4,&C3,&89,&C1,&C3,&DC,&C4,&C3,&33,&C5,&C
3,&6C,&C5,&C3,&93,&C5, 2814
40 DATA &C3,&29,&C6,&C3,&36,&C6,&C3,&43,&C6,&C3,&9
E,&C6,&C3,&EE,&C7,&C3, 2719
50 DATA &44,&C9,&C3,&75,&C9,&C3,&9B,&C9,&C3,&1C,&C
A,&C3,&2B,&CA,&C3,&3A, 2451
60 DATA &CA,&49,&4E,&49,&D4,&48,&4F,&43,&C8,&52,&5
5,&4E,&54,&45,&D2,&4E, 1742
70 DATA &45,&58,&54,&4C,&49,&4E,&C5,&50,&52,&45,&4
C,&49,&4E,&C5,&4E,&45, 1467
80 DATA &58,&54,&42,&59,&54,&C5,&50,&52,&45,&42,&5
9,&54,&C5,&52,&44,&43, 1492
90 DATA &48,&41,&D2,&46,&49,&4C,&CC,&54,&C2,&57,&5
2,&43,&41,&D3,&52,&44, 1710
100 DATA &43,&41,&D3,&44,&45,&CC,&53,&43,&52,&49,&
50,&D4,&4E,&4F,&CE,&4E, 1722
110 DATA &4F,&46,&C6,&48,&C3,&44,&55,&4D,&D0,&4D,&
4F,&44,&49,&46,&D9,&46, 1706
120 DATA &4C,&4F,&4F,&C4,&4D,&45,&44,&49,&D4,&53,&
45,&41,&52,&43,&C8,&47, 1566
130 DATA &45,&D4,&50,&55,&D4,&49,&4E,&46,&CF,&00,&
C5,&D5,&E5,&3E,&02,&CD, 1994
140 DATA &0E,&BC,&01,&00,&00,&CD,&38,&BC,&AF,&01,&
00,&00,&CD,&32,&BC,&21, 1304
150 DATA &CA,&C2,&CD,&8C,&C1,&CD,&2E,&C1,&3E,&FF,&
32,&00,&AC,&3A,&66,&BE, 2267
160 DATA &FE,&10,&28,&05,&37,&E1,&D1,&C1,&C9,&21,&
EA,&C1,&CD,&8C,&C1,&CD, 2401
170 DATA &95,&C1,&20,&08,&0E,&07,&21,&B2,&C1,&C3,&
16,&BD,&21,&13,&C2,&CD, 1664
180 DATA &8C,&C1,&CD,&95,&C1,&20,&12,&CD,&A0,&C1,&
21,&A4,&AC,&DF,&E7,&C1, 2504
190 DATA &CD,&AC,&C1,&21,&40,&C2,&CD,&8C,&C1,&E1,&
D1,&C1,&37,&C9,&21,&C0, 2507
200 DATA &C1,&0E,&06,&7E,&23,&46,&23,&E5,&CD,&2D,&
BB,&E1,&0D,&20,&F4,&21, 1692
210 DATA &CC,&C1,&11,&A4,&AC,&01,&1B,&00,&ED,&B0,&
06,&8D,&0E,&05,&21,&A4, 1554
220 DATA &AC,&CD,&0F,&BB,&06,&8E,&0E,&05,&21,&A9,&
AC,&CD,&0F,&BB,&06,&8F, 1676
230 DATA &0E,&04,&21,&AE,&AC,&CD,&0F,&BB,&06,&90,&
0E,&04,&21,&B2,&AC,&CD, 1560
240 DATA &0F,&BB,&06,&91,&0E,&05,&21,&B6,&AC,&CD,&
0F,&BB,&06,&92,&0E,&04, 1336
250 DATA &21,&BB,&AC,&CD,&0F,&BB,&CD,&AC,&C1,&21,&
5E,&C2,&7E,&B7,&C8,&CD, 2404
260 DATA &5A,&BB,&23,&18,&F7,&3E,&00,&32,&A4,&AC,&
CD,&18,&BB,&FE,&0D,&C9, 1915
270 DATA &21,&B7,&C1,&11,&A4,&AC,&01,&09,&00,&ED,&
B0,&C9,&AF,&21,&A4,&AC, 1930
280 DATA &06,&20,&77,&23,&10,&FC,&C9,&23,&00,&C8,&
00,&01,&01,&0A,&01,&03, 912
290 DATA &0F,&8D,&07,&8E,&06,&8F,&0D,&90,&0E,&91,&
05,&92,&4C,&49,&53,&54, 1237
300 DATA &20,&45,&44,&49,&54,&20,&52,&55,&4E,&0D,&
43,&41,&54,&0D,&41,&55, 995
310 DATA &54,&4F,&0D,&43,&4C,&53,&0D,&0D,&C6,&07,&
0A,&0D,&53,&6F,&6C,&6C, 1066
320 DATA &20,&43,&50,&2F,&4D,&20,&67,&65,&62,&6F,&
6F,&74,&65,&74,&20,&77, 1343
330 DATA &65,&72,&64,&65,&6E,&20,&3F,&20,&3C,&45,&
4E,&54,&45,&52,&3E,&20, 1189
340 DATA &0A,&0D,&00,&53,&6F,&6C,&6C,&20,&54,&55,&
52,&42,&4F,&2D,&44,&4F, 1053
350 DATA &53,&20,&61,&6B,&74,&69,&76,&69,&65,&72,&
74,&20,&77,&65,&72,&64, 1560
360 DATA &65,&6E,&20,&3F,&20,&3C,&45,&4E,&54,&45,&
52,&3E,&20,&0A,&0D,&00, 897
370 DATA &2D,&2D,&2D,&2D,&2D,&20,&54,&55,&52,&42,&
4F,&2D,&44,&4F,&53,&20, 960
380 DATA &41,&4B,&54,&49,&56,&20,&2D,&2D,&2D,&2D,&
2D,&0A,&0D,&00,&0A,&0D, 686
390 DATA &54,&61,&73,&74,&65,&6E,&62,&65,&6C,&65,&
67,&75,&6E,&67,&20,&69, 1601
400 DATA &6D,&20,&5A,&65,&68,&6E,&65,&72,&62,&6C,&
6F,&63,&6B,&20,&3C,&53, 1459
410 DATA &48,&49,&46,&54,&3E,&20,&3A,&0A,&0D,&3C,&
43,&41,&54,&63,&72,&3E, 1025
420 DATA &20,&20,&20,&20,&3C,&41,&55,&54,&4F,&63,&
72,&3E,&20,&20,&20,&20, 904
430 DATA &3C,&43,&4C,&53,&63,&72,&3E,&0A,&0D,&3C,&
4C,&49,&53,&54,&20,&3E, 1054
440 DATA &20,&20,&20,&20,&3C,&45,&44,&49,&54,&20,&
3E,&20,&20,&20,&20,&20, 736
450 DATA &3C,&52,&55,&4E,&63,&72,&3E,&0A,&0D,&00,&
00,&00,&00,&00,&00,&00, 603
460 DATA &00,&00,&00,&00,&00,&00,&00,&00,&00,&00,&
00,&00,&00,&00,&00,&00, 0
470 DATA &00,&00,&00,&00,&00,&00,&00,&00,&00,&00,&
00,&00,&00,&00,&00,&00, 0
480 DATA &00,&00,&00,&00,&00,&00,&00,&00,&00,&00,&
00,&00,&00,&00,&00,&00, 0
490 DATA &00,&00,&00,&00,&00,&00,&00,&00,&00,&00,&
00,&00,&00,&00,&00,&00, 0
500 DATA &00,&00,&00,&00,&00,&00,&00,&00,&00,&00,&
00,&00,&00,&00,&00,&00, 0
510 DATA &00,&00,&00,&00,&00,&00,&00,&00,&00,&00,&
00,&00,&00,&00,&00,&00, 0
520 DATA &00,&00,&00,&00,&00,&00,&00,&00,&00,&00,&
00,&00,&00,&00,&00,&00, 0
530 DATA &00,&00,&00,&00,&00,&00,&00,&00,&00,&00,&
00,&00,&00,&00,&00,&00, 0
540 DATA &00,&00,&00,&00,&00,&00,&00,&00,&00,&00,&
00,&00,&00,&00,&00,&00, 0
550 DATA &00,&00,&00,&00,&00,&00,&00,&00,&00,&00,&
00,&00,&00,&00,&00,&00, 0
560 DATA &00,&00,&00,&00,&00,&00,&00,&00,&00,&00,&
00,&00,&00,&00,&00,&00, 0
570 DATA &00,&00,&00,&00,&00,&00,&00,&00,&00,&00,&
00,&00,&00,&00,&00,&00, 0
580 DATA &00,&00,&00,&00,&00,&00,&00,&00,&00,&00,&
00,&00,&00,&00,&00,&00, 0
590 DATA &00,&00,&00,&00,&00,&00,&00,&00,&00,&00,&
00,&00,&00,&00,&00,&00, 0
600 DATA &00,&00,&00,&00,&00,&00,&00,&00,&00,&00,&
00,&00,&00,&00,&00,&00, 0
610 DATA &00,&00,&00,&00,&00,&00,&00,&00,&00,&00,&
FE,&01,&C2,&57,&C4,&DD, 953
620 DATA &7E,&00,&06,&01,&CD,&4D,&BC,&C9,&FE,&01,&
C2,&57,&C4,&DD,&7E,&00, 1883
630 DATA &06,&00,&CD,&4D,&BC,&C9,&FE,&01,&C2,&57,&
C4,&CD,&1E,&C4,&CD,&26, 2083
640 DATA &BC,&7C,&02,&0B,&7D,&02,&C9,&FE,&01,&C2,&
57,&C4,&CD,&1E,&C4,&CD, 2021
650 DATA &29,&BC,&18,&ED,&FE,&01,&C2,&57,&C4,&CD,&
1E,&C4,&CD,&20,&BC,&18, 2102
660 DATA &E0,&FE,&01,&C2,&57,&C4,&CD,&1E,&C4,&CD,&
23,&BC,&18,&D3,&DD,&46, 2341
670 DATA &01,&DD,&4E,&00,&0A,&6F,&03,&0A,&67,&E5,&
C5,&01,&00,&40,&09,&C1, 1230
680 DATA &E1,&D8,&E1,&21,&3A,&C4,&CD,&8C,&C1,&C9,&
0A,&0D,&55,&6E,&62,&72, 2122
690 DATA &61,&75,&63,&68,&62,&61,&72,&65,&20,&41,&
64,&72,&65,&73,&73,&65, 1570
700 DATA &20,&21,&21,&21,&0A,&0D,&00,&21,&5E,&C4,&
CD,&8C,&C1,&C9,&0A,&0D, 1239
710 DATA &46,&65,&68,&6C,&65,&6B,&64,&65,&72,&20,&
50,&61,&72,&61,&6D,&65, 1539
720 DATA &74,&65,&72,&20,&21,&21,&21,&0A,&0D,&00,&
FE,&01,&C2,&57,&C4,&DD, 1438
730 DATA &66,&01,&DD,&6E,&00,&CD,&60,&BB,&30,&02,&
77,&C9,&21,&93,&C4,&CD, 1873
740 DATA &8C,&C1,&C9,&0A,&0D,&4B,&65,&69,&6E,&20,&
72,&65,&67,&75,&6C,&61, 1620
750 DATA &65,&72,&65,&73,&20,&5A,&65,&69,&63,&68,&
65,&6E,&20,&21,&21,&21, 1304
760 DATA &0A,&0D,&00,&FE,&05,&C2,&57,&C4,&CD,&0B,&
BC,&47,&0E,&00,&09,&DD, 1478
770 DATA &46,&00,&CD,&26,&BC,&10,&FB,&DD,&46,&02,&
```

```
CD,&20,&BC,&10,&FB,&DD, 1974
780 DATA &56,&04,&DD,&5E,&06,&DD,&4E,&08,&CD,&47,&
BC,&C9,&FE,&04,&C2,&57, 1922
790 DATA &C4,&21,&16,&C5,&CD,&8C,&C1,&CD,&95,&C1,&
DD,&6E,&00,&DD,&66,&01, 2188
800 DATA &3E,&0A,&CD,&68,&BC,&CD,&6E,&BC,&DD,&6E,&
06,&DD,&66,&07,&DD,&5E, 2054
810 DATA &04,&DD,&56,&05,&DD,&7E,&02,&CD,&9E,&BC,&
CD,&71,&BC,&21,&50,&01, 1836
820 DATA &3E,&0A,&CD,&68,&BC,&C9,&0A,&0D,&3C,&52,&
45,&43,&3E,&20,&26,&20, 1235
830 DATA &3C,&50,&4C,&41,&59,&3E,&20,&26,&20,&3C,&
54,&41,&53,&54,&45,&3E, 1041
840 DATA &0A,&0D,&00,&FE,&02,&C2,&57,&C4,&21,&57,&
C5,&CD,&8C,&C1,&CD,&95, 1965
850 DATA &C1,&CD,&6E,&BC,&DD,&6E,&02,&DD,&66,&03,&
DD,&7E,&00,&11,&00,&00, 1719
860 DATA &CD,&A1,&BC,&CD,&71,&BC,&C9,&0A,&0D,&3C,&
50,&4C,&41,&59,&3E,&20, 1748
870 DATA &26,&20,&3C,&54,&41,&53,&54,&45,&3E,&0A,&
0D,&00,&FE,&01,&C2,&57, 1136
880 DATA &C4,&CD,&0B,&BC,&47,&0E,&00,&09,&E5,&0E,&
C8,&CD,&19,&BD,&06,&50, 1642
890 DATA &DD,&7E,&00,&77,&CD,&20,&BC,&10,&F7,&E1,&
CD,&26,&BC,&E5,&0D,&20, 2084
900 DATA &EA,&E1,&C9,&FE,&01,&28,&04,&DD,&21,&07,&
00,&3E,&02,&CD,&0E,&BC, 1691
910 DATA &01,&04,&04,&CD,&38,&BC,&21,&DE,&C5,&CD,&
8C,&C1,&21,&A4,&AC,&06, 1823
920 DATA &FF,&AF,&77,&23,&10,&FC,&21,&A4,&AC,&CD,&
3A,&BD,&3A,&A4,&AC,&B7, 2250
930 DATA &28,&15,&21,&A4,&AC,&DD,&46,&00,&CD,&E7,&
C5,&3E,&0D,&CD,&5A,&BB, 1911
940 DATA &3E,&0A,&CD,&5A,&BB,&18,&CF,&01,&00,&00,&
CD,&38,&BC,&C9,&09,&09, 1454
950 DATA &09,&09,&00,&00,&00,&00,&00,&CD,&2E,&BD,&
30,&07,&21,&13,&C6,&CD, 968
960 DATA &8C,&C1,&C9,&78,&E6,&0F,&47,&3E,&20,&CD,&
89,&C6,&10,&F9,&7E,&B7, 2178
970 DATA &28,&06,&CD,&89,&C6,&23,&18,&F6,&3E,&0A,&
CD,&89,&C6,&3E,&0D,&CD, 1783
980 DATA &89,&C6,&C9,&0A,&0D,&44,&52,&55,&43,&4B,&
45,&52,&20,&46,&45,&48, 1330
990 DATA &4C,&54,&20,&21,&21,&21,&0A,&0D,&00,&21,&
32,&C6,&06,&00,&CD,&E7, 1037
1000 DATA &C5,&C9,&1B,&49,&03,&00,&21,&3F,&C6,&06,
&00,&CD,&E7,&C5,&C9,&1B, 1662
1010 DATA &49,&01,&00,&CD,&2E,&BD,&30,&07,&21,&13,
&C6,&CD,&8C,&C1,&C9,&3E, 1620
1020 DATA &0A,&CD,&89,&C6,&3E,&0D,&CD,&89,&C6,&2E,
&01,&26,&01,&E5,&CD,&75, 1802
1030 DATA &BB,&CD,&60,&BB,&38,&02,&3E,&20,&CD,&89,
&C6,&E1,&CD,&92,&C6,&C0, 2333
1040 DATA &24,&7C,&FE,&51,&20,&E7,&3E,&0A,&CD,&89,
&C6,&3E,&0D,&CD,&89,&C6, 1985
1050 DATA &26,&01,&2C,&7D,&FE,&1A,&C8,&18,&D4,&CD,
&2E,&BD,&38,&FB,&CD,&31, 1925
1060 DATA &BD,&C9,&E5,&C5,&D5,&3E,&42,&CD,&1E,&BB,
&D1,&C1,&E1,&C9,&FD,&21, 2693
1070 DATA &00,&00,&FE,&03,&20,&0B,&CD,&2E,&BD,&38,
&1E,&FD,&21,&01,&00,&18, 1137
1080 DATA &18,&FE,&02,&28,&33,&FE,&01,&C2,&57,&C4,
&DD,&6E,&00,&DD,&66,&01, 1758
1090 DATA &7D,&E6,&F0,&6F,&5D,&54,&14,&18,&2D,&DD,
&6E,&04,&DD,&66,&05,&DD, 1856
1100 DATA &5E,&02,&DD,&56,&03,&E5,&37,&3F,&ED,&52,
&E1,&D2,&58,&C7,&7D,&E6, 2149
1110 DATA &F0,&6F,&7B,&F6,&0F,&5F,&18,&0E,&DD,&6E,
&02,&DD,&66,&03,&DD,&5E, 1842
1120 DATA &00,&DD,&56,&01,&18,&DF,&E5,&D5,&3E,&02,
&CD,&0E,&BC,&21,&75,&C7, 1817
1130 DATA &CD,&8C,&C1,&D1,&E1,&7C,&CD,&B7,&C7,&7D,
&CD,&B7,&C7,&3E,&20,&CD, 2694
1140 DATA &9F,&C7,&3E,&20,&CD,&9F,&C7,&06,&10,&E5,
&E7,&CD,&B7,&C7,&23,&3E, 2181
1150 DATA &20,&CD,&9F,&C7,&10,&F4,&E1,&3E,&20,&CD,
&9F,&C7,&3E,&20,&CD,&9F, 2195
1160 DATA &C7,&06,&10,&E7,&CD,&9F,&C7,&23,&10,&F9,
&3E,&0A,&CD,&A7,&C7,&3E, 2020
1170 DATA &0D,&CD,&A7,&C7,&E5,&C5,&3E,&15,&CD,&1E,
&BB,&20,&F9,&C1,&E1,&E5, 2443
1180 DATA &37,&3F,&ED,&52,&E1,&D0,&18,&AD,&21,&5F,
&C7,&CD,&8C,&C1,&C9,&0A, 2143
1190 DATA &0D,&41,&6E,&66,&61,&6E,&67,&20,&3E,&20,
&45,&6E,&64,&65,&20,&21, 1171
1200 DATA &21,&21,&0A,&0D,&00,&52,&41,&4D,&20,&48,
&65,&78,&64,&75,&6D,&70, 1076
1210 DATA &20,&3A,&20,&41,&64,&72,&65,&73,&73,&65,
&20,&2F,&20,&42,&79,&74, 1247
1220 DATA &65,&73,&20,&2F,&20,&41,&53,&43,&49,&49,
&0A,&0D,&0A,&0D,&00,&E6, 964
1230 DATA &7F,&FE,&20,&30,&02,&3E,&2E,&F5,&3E,&89,
&FD,&BE,&00,&28,&04,&F1, 1743
1240 DATA &C3,&5A,&BB,&F1,&C3,&89,&C6,&E5,&C5,&F5,
&E6,&F0,&0F,&0F,&0F,&0F, 2444
1250 DATA &21,&DC,&C7,&4F,&06,&00,&09,&7E,&CD,&9F,
&C7,&F1,&E6,&0F,&21,&DC, 1974
1260 DATA &C7,&4F,&06,&00,&09,&7E,&CD,&9F,&C7,&C1,
&E1,&C9,&30,&31,&32,&33, 1799
1270 DATA &34,&35,&36,&37,&38,&39,&41,&42,&43,&44,
&45,&46,&00,&00,&FE,&01, 987
1280 DATA &C2,&57,&C4,&FD,&21,&00,&00,&DD,&6E,&00,
&DD,&66,&01,&01,&70,&01, 1532
1290 DATA &E5,&37,&3F,&ED,&42,&E1,&30,&06,&21,&E0,
&C8,&C3,&8C,&C1,&E5,&3E, 2205
1300 DATA &02,&CD,&0E,&BC,&3E,&FF,&01,&E8,&B4,&02,
&21,&F9,&C8,&CD,&8C,&C1, 2161
1310 DATA &E1,&7C,&CD,&B7,&C7,&7D,&CD,&B7,&C7,&06,
&06,&3E,&90,&CD,&5A,&BB, 2348
1320 DATA &10,&F9,&E7,&CD,&B7,&C7,&06,&08,&3E,&90,
&CD,&5A,&BB,&10,&F9,&CD, 2255
1330 DATA &82,&C8,&FE,&0D,&20,&11,&E7,&CD,&B7,&C7,
&23,&3E,&0A,&CD,&5A,&BB, 2053
1340 DATA &3E,&0D,&CD,&5A,&BB,&18,&CA,&FE,&58,&28,
&21,&57,&CD,&82,&C8,&5F, 1915
1350 DATA &CD,&98,&C8,&7A,&CD,&5A,&BB,&7B,&CD,&5A,
&BB,&CD,&BC,&C8,&77,&23, 2513
1360 DATA &3E,&0A,&CD,&5A,&BB,&3E,&0D,&CD,&5A,&BB,
&18,&A5,&01,&E8,&B4,&AF, 1888
1370 DATA &02,&C9,&E5,&D5,&CD,&81,&BB,&CD,&7B,&BB,
&CD,&18,&BB,&F5,&CD,&7E, 2673
1380 DATA &BB,&CD,&84,&BB,&F1,&D1,&E1,&C9,&E5,&21,
&DC,&C7,&7B,&CD,&B0,&C8, 2972
1390 DATA &28,&02,&1E,&30,&21,&DC,&C7,&7A,&CD,&B0,
&C8,&E1,&C8,&16,&30,&C9, 1971
1400 DATA &06,&10,&BE,&C8,&23,&05,&20,&FA,&3E,&FF,
&B7,&C9,&E5,&21,&DC,&C7, 2116
1410 DATA &7B,&CD,&D8,&C8,&58,&21,&DC,&C7,&7A,&CD,
&D8,&C8,&50,&7A,&07,&07, 2243
1420 DATA &07,&07,&57,&AF,&B2,&B3,&E1,&C9,&06,&00,
&BE,&C8,&23,&04,&18,&FA, 1768
1430 DATA &0A,&0D,&41,&64,&72,&65,&73,&73,&65,&20,
&7A,&75,&20,&6B,&6C,&65, 1353
1440 DATA &69,&6E,&20,&21,&21,&21,&0A,&0D,&00,&52,
&41,&4D,&20,&4D,&6F,&64, 913
1450 DATA &69,&66,&69,&6B,&61,&74,&69,&6F,&6E,&20,
&3A,&20,&41,&64,&72,&65, 1460
1460 DATA &73,&73,&65,&20,&2F,&20,&41,&6C,&74,&20,
&2F,&20,&4E,&65,&75,&0A, 1148
1470 DATA &0D,&3C,&58,&3E,&20,&3A,&20,&45,&6E,&64,
&65,&20,&2F,&20,&3C,&45, 965
1480 DATA &6E,&74,&65,&72,&3E,&20,&3A,&20,&41,&6C,
&74,&3D,&4E,&65,&75,&0A, 1281
1490 DATA &0D,&0A,&0D,&00,&FE,&03,&C2,&57,&C4,&DD,
&6E,&04,&DD,&66,&05,&E5, 1662
1500 DATA &01,&70,&01,&37,&3F,&ED,&42,&E1,&30,&06,
&21,&E0,&C8,&C3,&8C,&C1, 1799
1510 DATA &DD,&4E,&02,&DD,&46,&03,&DD,&7E,&00,&77,
&23,&0B,&78,&FE,&00,&20, 1513
1520 DATA &F5,&B9,&C8,&18,&F1,&FE,&01,&C2,&57,&C4,
&21,&A4,&AC,&06,&FF,&AF, 2432
1530 DATA &77,&23,&10,&FC,&21,&A4,&AC,&CD,&3A,&BD,
&21,&A4,&AC,&DD,&5E,&00, 1927
1540 DATA &DD,&56,&01,&7E,&12,&B7,&C8,&13,&23,&18,
&F8,&FE,&03,&C2,&57,&C4, 1895
1550 DATA &3E,&02,&CD,&0E,&BC,&21,&E0,&C9,&CD,&8C,
&C1,&FD,&21,&00,&00,&CD, 1958
1560 DATA &09,&B9,&DD,&6E,&04,&DD,&66,&05,&DD,&4E,
&02,&DD,&46,&03,&DD,&7E, 1799
1570 DATA &00,&ED,&A1,&CC,&CF,&C9,&78,&FE,&00,&20,
&F3,&B9,&C8,&18,&EF,&C5, 2504
1580 DATA &E5,&2B,&11,&0F,&00,&19,&EB,&E1,&E5,&2B,
&CD,&05,&C7,&E1,&C1,&C9, 2089
1590 DATA &53,&75,&63,&68,&65,&20,&69,&6D,&20,&52,
&41,&4D,&20,&3A,&20,&41, 1193
1600 DATA &64,&72,&65,&73,&73,&65,&20,&2F,&20,&42,
&79,&74,&65,&73,&20,&2F, 1355
1610 DATA &20,&41,&53,&43,&49,&49,&20,&0A,&0D,&0A,
&0D,&00,&DD,&6E,&00,&DD, 1023
1620 DATA &66,&01,&DD,&4E,&02,&DD,&56,&04,&DD,&5E,
&06,&C9,&FE,&04,&C2,&57, 1776
1630 DATA &C4,&CD,&0C,&CA,&DF,&28,&CA,&C9,&66,&C6,
&07,&FE,&04,&C2,&57,&C4, 2323
1640 DATA &CD,&0C,&CA,&DF,&37,&CA,&C9,&4E,&C6,&07,
&C9,&62,&CA,&E5,&3E,&02, 2177
1650 DATA &CD,&0E,&BC,&E1,&CD,&4E,&CA,&E5,&CD,&18,
&BB,&E1,&18,&EF,&06,&18, 2280
1660 DATA &7E,&FE,&0D,&20,&02,&05,&C8,&B7,&20,&02,
&E1,&C9,&CD,&5A,&BB,&23, 1792
1670 DATA &18,&EE,&20,&20,&20,&20,&20,&20,&20,&20,
&20,&20,&20,&20,&20,&20, 710
1680 DATA &20,&20,&20,&20,&20,&20,&20,&20,&20,&20,
&20,&44,&69,&65,&20,&42, 724
1690 DATA &41,&53,&49,&43,&20,&2D,&20,&45,&72,&77,
&65,&69,&74,&65,&72,&75, 1353
1700 DATA &6E,&57,&20,&52,&4F,&4D,&53,&4F,&46,&54,
&0D,&0A,&0D,&0A,&20,&20, 909
1710 DATA &20,&20,&20,&20,&20,&00,&00,&00,&00,&00,
&00,&00,&00,&00,&00,&00, 160
1720 DATA &00, 0
1730 dat=0 : sz=0 : dz = 10
1740 FOR adr = 16384  TO 19120
1750   READ byte : dat=dat+1
1760   sz=sz+byte
1770   POKE adr,byte
1780   IF dat < 16 AND adr <  19120 THEN 1820
1790   READ chksum
1800   IF chksum<>sz THEN PRINT "Fehler in Zeile :
";dz
1810   dz=dz + 10 : sz=0 : dat=0
1820 NEXT adr
1830 END
```

CPC 464, 664 + 6128 Software

STAR-WRITER I

Das zur Zeit mit Abstand leistungsfähigste Textsystem für den CPC, das völlig neue Maßstäbe in Bedienung und Leistung setzt!

- Einfache Bedienung durch PULL-DOWN Menues
- Deutsche DIN-Tastatur mit Aufkleber
- Integrierte Adreßverwaltung im Direktzugriff
- Integriertes Grafikprogramm zur Erstellung von Briefköpfen
- Integriertes DFÜ-Programm für die Schneider-Schnittstelle
- Integrierter Zeicheneditor zur Erstellung von 10 Zeichensätzen
- Anpaßbar an jeden Drucker der auf dem Markt erhältlich ist
- Integrierte Tabellenkalkulation im Textprogramm
- Basicprogramme können eingelesen und bearbeitet werden
- Komfortable Diskettenverwaltung (löschen, kopieren, umbenennen...)
- Ränder frei wählbar
- Trennvorschläge
- Zentrieren
- Formatierte Ausgabe auf dem Bildschirm
- Umfangreiche Blockoperationen
- Umformatieren auf andere Formate
- Zeichen einfügen und löschen
- Zeilen einfügen und löschen
- Absätze einfügen und löschen
- Kopf- und Fußzeilen frei änderbar
- Blocksatz
- Flattersatz
- Suchen und Ersetzen
- Serienbrieffunktion
- „MERGE"-Funktion
- Fließtexteingabe
- Wordwrap



Jetzt Version 2.0

Achtung an alle Vortex-Anwender!

Ab sofort alle Programme auf 5¼ Zoll Disketten unter VDOS 2.0 lieferbar.
Die hohe Speicherkapazität und alle anderen Vorzüge des Laufwerks werden von den Programmen voll ausgenutzt!

- Bausteinverarbeitung
- 9 verschiedene Layouts
- Proportionalschrift
- Hervorhebungen wie: Unterstreichen, Fettdruck, Doppeldruck, Kursiv, doppelte Breite, Schmalschrift, NLQ...
- Mathematische Formelschreibweise (Hoch- und Tiefstellen)
- Definierte Zeichensätze können auf entsprechendem Drucker ausgegeben werden (z. B. Russisch oder Französisch).

Software des Jahres Platz 3

STAR-WRITER I, ein Textsystem der Superlative, wurde schon nach einem Monat Marktanwesenheit bei der Wahl zur ***Software des Jahres*** auf den dritten Platz gewählt. Diese Wahl bestätigt den Anspruch von STAR-WRITER I, ein Textsystem der Superlative zu sein, völlig. Sollten Sie noch an STAR-WRITER I zweifeln, fordern Sie unser kostenloses Informationsmaterial an.

COPY-STAR II

ist die ideale Befehlserweiterung für Druckerbesitzer, denn es stellt für alle gängigen Drucker Hardcopyfunktionen in verschiedenen Größen zur Verfügung. Sogar Farbbilder lassen sich schattiert ausgeben. COPY-STAR II können Sie leicht in eigene Programme einbinden.

Preis: ***29,90*** Kassette
39,90 Diskette

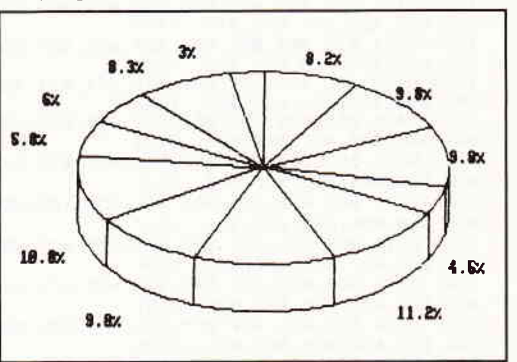
STATISTIC-STAR

Eine Grafik sagt mehr als 1000 Zahlen

Ein professionelles Grafik- und Statisticprogramm zum Auswerten von Daten aller Art (Schule, Studium, Beruf, Hobby, Haushalt...).

- Linien-, Balken- und Tortengrafik
- Betiteln von Grafiken
- 400 Daten direkt im Speicher
- Umfangreiche Editierfunktionen
- Umfangreiche statistische Berechnungen
- Hardcopyfunktion u.v.m.
- Preis: ***59,90*** Kassette
79,90 Diskette

Beispielgrafik STATISTIC-STAR



MATHE-STAR

Vom Lehrer für Schüler!

- Lin. Gleichungssysteme
- Gleichungen 4. Grades
- Bruchrechnen
- Primfaktorzerlegung
- Polynome
- Kurvendiskussion
- Integralrechnung
- Vektorrechnung
- Matrixrechnung
etc.

Preis: ***69,90*** Kassette
79,90 Diskette

DISKSORT-STAR

Leistungsstarke Diskettenverwaltung, die keinem CPC-Benutzer fehlen sollte. DISKSORT-STAR verwaltet, archiviert, katalogisiert, druckt... Ihre Diskettensammlung auf einfachste Weise. Neben der reinen Diskettenverwaltung ist unter anderem noch ein kompletter Diskettenmanager enthalten!
Auch in punkto Bedienung ist DISKSORT-STAR kaum zu schlagen

DM 49,90

STAR-MON

Das Entwicklungssystem für Profis und Anfänger

- Assembler
- Editor
- Disassembler
- Monitor
- 4 Breakpoints
- Trace-Funktion
- Bankswitch
- Memory Dump
- Diskettenmonitor
- u.v.m.

Preis: ***59,90*** Kassette
79,90 Diskette
(erweiterte Version)

CREATOR-STAR

Ein Trickfilmdesigner für Ihren CPC und alle Hobbyregisseure!

- Sprite-Designer
- Laufschrift
- Utilities
- Kulissendesigner
- Sprites mit 4 Unterpositionen
- Verbinden von Sprites
- Kulissen auch übereinandergelegt
- Eigene Programmiersprache mit Editor und Compiler

DM 49,90 Diskette

JOYCE + JOYCE PLUS Software



DATEI STAR (464 / 664 / 6128 / JOYCE)

Das universelle Dateiverwaltungsprogramm im Direktzugriff!

- Einfache Bedienung durch PULL-DOWN Menues
- Frei definierbare Eingabemaske mit 30 Datenfeldern
- 1400 Zeichen pro Datensatz!
- Frei definierbare Eingabe-, Such- und Druckmaske
- Suchen und Selektieren nach allen Feldern gleichzeitig
- Umfangreiche Sortierfunktionen und Erstellung einer Pointerdatei!
- Integrierte Kalkulationsmöglichkeiten (Rechenfelder)
- Schnittstelle zu STAR-WRITER I (nur CPC)
- Integrierter Reportgenerator u. v. m.
- Preis ***98,— DM***

STAR-MAIL (nur für JOYCE)

Die Schnittstelle zur Außenwelt für alle JOYCE-Benutzer!

- Automatische Serienbrieferstellung in Verbindung mit Locoscript und Datei-Star. Alle Daten aus DATEI-STAR können beliebig in Locoscript-Texte eingebaut und verwendet werden.
- Endlich können Locoscript-Texte mit fremden Druckern, sogar Typenradschreibmaschinen, verarbeitet werden. Hierbei ist zu beachten, daß original Locoscript-Texte mit allen Steuerzeichen verwendet werden, und nicht wie bei anderen Programmen umgewandelte ASCII-Dateien!
- Komplette Unterstützung der Centronics- & RS 232 Schnittstelle
- Jeder Drucker kann mit Hilfe einer einfachen Anpassung verwendet werden!
- **DATEI-STAR & STAR-MAIL gibt es im Paket als JOYCE-MAILING-SYSTEM für nur 189,— DM.**
- Preis: ***98,— DM***

FIBU-STAR Plus

(Joyce, 6128 oder 664 und 464 mit Speichererw.)

Endlich gibt es eine professionelle Finanzbuchhaltung für die SCHNEIDER-Computer!

- Stammdatenverwaltung und Umsatzsteuervoranmeldung
- Frei wählbarer Umsatzsteuerschlüssel zur automatischen Ermittlung der Steuerbeträge und deren Verbuchung auf den Mehrwert- und Vorsteuerkonten
- Frei wählbarer Kontenrahmen (bis zu 400 Konten)
- Frei wählbare Zuordnung der Sach- und rechnenden Konten
- Buchungsjournal, Kontenblätter und Jahresabschluß
- Summen- und Saldenlisten der Kreditoren, Debitoren und Sachkonten
- Einfaches Erfassen aller Buchungsbelege mit oder ohne Protokoll
- Buchungen können so durchgeführt werden, wie sie auf den Tisch kommen, da alle Buchungen chronologisch und innerhalb eines Datums nach Belegnummern sortiert werden
- Einfache Bedienung und hohe ABSICHERUNG gegen Bedienungsfehler
- Preis ***298,— DM***

Das Programm FIBU-STAR Plus ist erstmals unter dem Namen FIBU-M 1980 installiert worden und hat sich seitdem sehr oft in der Praxis bewährt.

STAR-BASE

STAR-BASE ist ein anwenderfreundliches Datenbanksystem mit unglaublichen Fähigkeiten!

- Eigener Window-Manager
- Menueorientierte Bedienung und Programmierung
- Volle Unterstützung des JOYCE-Bildschirms
- Eingabemaske über 9 Bildschirmseiten
- 100 Datenfelder pro Eingabemaske
- 88 Zeichen pro Datenfeld
- 2060 Zeichen pro Datensatz
- Max. 8 Zugriffsschlüssel
- Komplette Indexverarbeitung
- Suchen u. Selektieren nach allen Datenfeldern
- 32767 Datensätze pro Datei
- Verknüpfung von mehreren Dateien
- Zu jedem Datensatz kann eine Unterdatei mit max. 255 Zeichen pro Satz angelegt werden
- Kompletter Reportgenerator
- Star-Base arbeitet voll mit Locoscript zusammen
- Umfangreiche Rechenfunktionen
- Serienbrieferstellung
- Integrierter Texteditor
- Direkter, menuegesteuerter Datensatzzugriff
- Mehrspaltige Formulare
- Eigener Menuegenerator
- Einfachste Maskendefinition
- Passwort für eine Datei
- Ausführliches Handbuch
- Preis: ***198,— DM*** inkl. Diskette u. Handbuch

Weitere Informationen zu STAR-BASE und zu unseren anderen JOYCE-Programmen finden Sie in unserem neuen JOYCE-Katalog!

Weitere interessante Programme finden Sie in unserem ausführlichen Katalog!

UELZENER STR. 12
2120 LÜNEBURG
FERNRUF (0 41 31) 40 25 50

STAR DIVISION

Vertrieb für die Schweiz:
VCS Video-Computershop
Schaffhauserstraße 473
P.O. Box 103, CH-8052 Zürich
Telefon 01 / 302 26 00

Vertrieb für Österreich:
Ueberreuter-Media
Alserstraße 24, A-1091 Wien
Telefon 481 53 80

Händleranfrage erwünscht

Software-Autoren gesucht

Unsere Produkte sind in allen führenden Computer-Shops sowie bei Karstadt und Brinkmann erhältlich!

☐ Senden Sie mir unverbindlich Ihren Katalog
☐ 6128 ☐ 664 ☐ 464 ☐ JOYCE
☐ Hiermit bestelle ich per Vorkasse/Nachnahme

Vorname ____________
Name ____________
Straße ____________
Wohnort ____________
Datum/Unterschrift ____________

(Alle Preise verstehen sich als unverbindliche Preisempfehlungen zzg. 5.— DM Porto und Verpackung)

CPC-D-786

Pascal

Zum Schluß des letzten Teiles dieses Kurses äußerte ich, daß noch ein wenig mehr zu den skalaren Datentypen zu sagen wäre, ich aus Platzmangel aber darauf verzichten mußte. Deshalb hier – bevor es mit Ausdrücken und Anweisungen weiter geht, der vorläufige Abschluß dieser Sektion.

Aus diesem letzten Teil kennen wir also die Pascal zugrunde liegenden skalaren Datentypen INTEGER, REAL, CHARacter und BOOLEAN. Ebenso die mit diesen Typen möglichen Operationen. Wie wir schon aus Teil 2 wissen, muß eine jede Größe in Pascal, bevor sie das erstemal in einem Programm verwendet wird, definiert werden. In den zwei Beispielprogrammen des 2. Teiles sind auch einfache Variablen-Vereinbarungen enthalten. Doch betrachten wir uns die Definition von Datentypen noch mal kurz genauer: Bei der Verwendung von Konstanten im Anweisungsteil oder der Deklaration derselben (CONST) im Vereinbarungsteil ergibt sich der Typ einer Konstanten implizit durch den angegebenen Wert. So handelt es sich z.B. bei der Zahl 4321 um eine INTEGER-Zahl, 0.14 ist eine REAL-Zahl und 'XYZ123?' ist eine Zeichenketten-Konstante.

Bei der Deklaration von VARiablen wird der Datentyp explizit angegeben. Der Compiler kann ja schließlich nicht erahnen, welchen Typ die Daten besitzen, die der Programmierer später in diese Variablen packen will. In den Beispielen wurden mehrere Variablen des gleichen Typs deklariert, indem sie durch Kommas getrennt aufgeführt wurden. Der letzten Variablen folgte dann ein Doppelpunkt und der gewünschte Datentyp. Man hätte aber auch jede einzelne Variable für sich deklarieren können. Dies sei hier erwähnt, da je nach Stil und Übersichtlichkeit die eine oder andere Form bevorzugt wird. Ebenso ist die Reihenfolge gleichgültig, mit der die Variablen mit den unterschiedlichen Typen definiert werden.

Zusätzlich zur Verwendung der vorgegebenen Standard-Datentypen besteht noch die Möglichkeit, für diese Datentypen andere Namen zu verwenden oder gänzlich neue Datentypen zu 'kreieren'. Die richtige Verwendung solcher benutzerdefinierter Datentypen kann sehr zur Selbstdokumentation von Programmen beitragen. Zum Beispiel könnte man in einem Programm zur Auswertung von Leistungen irgendeiner Art (Sport, Schule etc.), bei denen zur Bewertung Punkte vergeben werden, variable Größen wie *gesamt, bonus, straf* einfach als INTEGER-Größen deklarieren:

```
VAR gesamt,bonus,straf:INTEGER;
```

Alternativ, und auch bei kleineren Programmen ratsam, kann man aber einen neuen Datentyp einführen – den Typ *punkte:*

```
TYPE punkte = INTEGER;
```

und die Variablen mit diesem Typ definieren:

```
VAR gesamt,bonus,straf:punkte;
```

Im ersten Moment mag es zwar wenig sinnvoll erscheinen, einen existierenden Datentyp einfach umzubenennen. Allerdings erlaubt Pascal hier, wie gesagt, noch wesentlich mehr. Aber auch dieses kleine Beispiel hilft bei der Programmerstellung und -dokumentation.

Besonders hilfreich wird es, wenn der neue Datentyp nur einen gewissen Wertebereich, also eine Teilmenge eines Datentyps oder ein paar bestimmte Werte umfaßt. Zum Beispiel könnte der Wertebereich des Datentypes *punkte* mit:

```
TYPE punkte = -100..100;
```

(lies: -100 bis 100) eingeschränkt werden. Eine Unter- oder Überschreitung dieses Teilbereiches bei Operationen mit Variablen des Typs *punkte* würde zu einem Fehler führen. Ein anderes Beispiel:

```
TYPE tage = 1..31;
```

Hierbei können Variablen des Types *tage* keinen Wert kleiner 1 und größer 31 annehmen. Unsinnige Werte (z.B.: 35. Septober) können nicht auftreten, da Pascal die Einhaltung solcher Teilbereiche überwacht.

Reihenfolge:

Bei den meisten Pascal-Implementationen muß die Vereinbarung der Konstanten, vom Benutzer definierte Datentypen und Variablen in einer festen Reihenfolge vorgenommen werden. Als erstes müssen die Konstanten definiert werden, falls notwendig. Das Schlüsselwort CONST leitet dieses ein und der Compiler erwartet solange Konstantendefinitionen der Art:

```
konstname = wert;
```

bis die Konstantenvereinbarung durch ein anderes Schlüsselwort aufgehoben wird. Danach können die eigenen Datentypen definiert werden, eingeleitet durch TYPE. Als vorerst letztes im Vereinbarungsteil werden dann die nötigen Variablen deklariert, eingeleitet mit dem Schlüsselwort VAR.

Baukastenprinzip...

Doch wenden wir uns nun einem zur Verarbeitung der kennengelernten Datentypen ohne Zweifel ebenso wichtigen Teil von Pascal zu: dem 'Arbeitstier' bzw. dem 'Anweisungsteil' oder 'Programmkörper.

In diesem sollen die, aus welchen Gründen auch immer, angefallenen Daten mit Hilfe von Operationen so verarbeitet/manipuliert werden, daß als Ergebnis auch etwas Brauchbares herauskommt. Wie die Bezeichnung 'Anweisungsteil' treffend ausdrückt, befinden sich hier die zur Verarbeitung der Daten und zur Steuerung des Programmes nötigen Anweisungen. Fangen wir aber bei der elementarsten Einheit des Anweisungteiles an, da diese in jedem (sinnvollen!?) Programm ausgiebig auftreten und gut verstanden werden wollen:

... die Ausdrücke.

Aus der Mathematik kennen wir diese bzw. so ähnlich Ausdrücke schon. Ganz einfach beschrieben, bestehen Ausdrücke aus einer Reihe von Elementen (Daten), die durch Operatoren (zur Erinnerung s. Teil 3) verknüpft werden:

```
-2
1+2+3+4
4711*123/x
a+b-c/2
```

Es sei noch einmal darauf hingewiesen, daß in Pascal die mit einem Datentyp ausführbaren Operationen von diesem Typ abhängig sind. Für einen Ausdruck ergibt sich daraus automatisch, daß in ihm nur Daten des gleichen Typs enthalten sein dürfen. So können auf Grund der Standardtypen für z.B. arithmetische Ausdrücke nur zwei Arten auftreten:

Ganzzahlige Ausdrücke, in denen nur ganze Zahlen benutzt werden dürfen, und reelle Ausdrücke, in denen entsprechend nur reelle Zahlen auftreten dürfen. Bei letzteren gibt es allerdings eine Ausnahme – es dürfen auch ganze Zahlen vorkommen, da diese bei Bedarf in reelle Zahlen umgewandelt werden (z.B. wird aus 2 die reelle Zahl 2.0).

Bleiben wir noch ein wenig bei den arithmetischen Ausdrücken. Ebenso ist die Regel "Punktrechnung geht vor Strichrechnung" und die Mög-

lichkeit des "Klammerns" aus der Mathematik bekannt. Diese Regeln befolgt auch Pascal bei der Auswertung von Ausdrücken:

8-2*3

wird folgendermaßen ausgewertet:

1. 2*3 = 6, also zuerst Punktrechnung.
2. 8-6 = 2, dann Strichrechnung.

oder

8+5-2*3

Auswertung:

1. 2*3 = 6, wie oben

Der Ausdruck lautet nun:

8+5-6

und enthält nur noch Operatoren 'gleicher Priorität'. Solche Ausdrücke werden nun einfach von links nach rechts ausgewertet:

2. 8+5 = 13
3. 13-6 = 7, fertig!

Und wenn man die Punkt-vor-Strich-Regel umgehen muß? Klammern!

(8+5-2)*3

Was kommt raus? 33?

Ausdrücke und Funktionen

Bis jetzt haben wir in unseren Ausdrücken nur Konstanten, Variablen und die nötigen Operatoren benutzt. Was ist aber mit den Standardfunktionen des letzten Teils? Sehen wir uns dies anhand folgenden Ausdruckes an:

$$ax^2+bx+c$$

Mit dem jetzigen Stand unseres Wissens kann man diesen Ausdruck in Pascal so schreiben:

a*(x*x)+b*x+c

Warum die Klammern? Da Punktrechnungen vor Strichrechnungen ausgeführt werden, werden bei weggelassenen Klammern zuerst die zwei Teilausdrücke:

1) a*x*x

und

2) b*x

ausgewertet, da die Multiplikationsoperatoren die größere Vorrangigkeit gegenüber den Additionsoperatoren besitzen. Die Ergebnisse der Teilausdrücke werden dann zu c addiert. Den erste Teilausdruck wertet Pascal wegen den gleichrangigen Operatoren von links nach rechts aus. Also zuerst:

a*x

Das daraus resultierende Ergebnis wird dann nochmal mit x multipliziert. Dies entspricht aber keinesfalls dem Ausdruck ax^2!

Durch die Klammern wird dies korrigiert: zuerst

x*x (=x^2)

und dann das Ergebnis mal a.

Nun stellt Pascal aber die Standardfunktion SQR(x) zur Verfügung, welche das Quadrat des Arguments x ermittelt. Also ersetzen wir unseren Klammerausdruck und schreiben einfach:

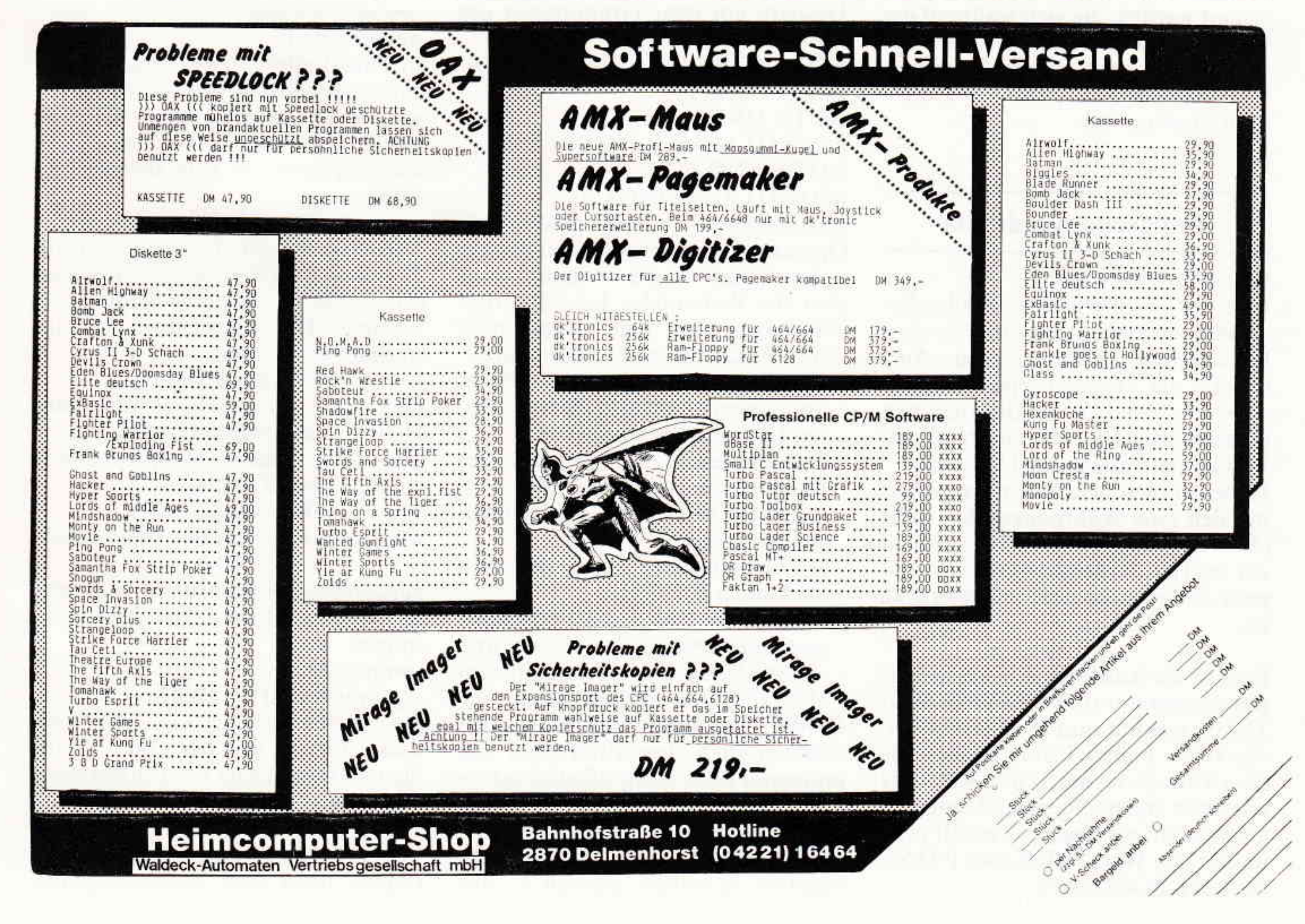
a*SQR(x)+b*x+c

Wie wir sehen, kann man Funktionen in einem Ausdruck überall dort einsetzen, wo sonst eine Konstante oder Variable steht. Allerdings ist dabei einiges zu beachten:

Das Ergebnis, welches die Funktion liefert, muß zum Datentyp der Konstanten oder Variablen, an deren Stelle sie eingesetzt wurde, passen. Befindet sich die Funktion zum Beispiel in einem ganzzahligen Ausdruck, so muß sie auch ein ganzzahliges Ergebnis liefern! Ebenso ist auf den Datentyp des Funktionargumentes (Operand) zu achten, wobei der Datentyp des Funktionargumentes nicht unbedingt der gleiche Datentyp wie der des Ergebnisses sein muß.

Beispiel:

Sei die Variable x als Integer-Größe vereinbart. Der Aufruf der Standardfunktion Sinus kann mit einem ganzzahligen oder reellen Wert als



Funktionsargument geschehen. Bei sin(x) in diesem Beispiel also mit einer ganzen Zahl. Der Datentyp des Ergebnisses dieser Funktion ist aber auf jeden Fall eine reelle Zahl. Die Funktion kann also nur in einem reellen Ausdruck verwendet werden. Soll sie aber doch in einem ganzzahligen Ausdruck zum Zuge kommen, so muß ihr Ergebnis irgendwie in eine ganze Zahl umgewandelt werden – zum Beispiel mit der ROUND-Funktion. Diese hat als Argument immer einen reellen Wert und liefert den ganzzahlig gerundeten Wert als Ergebnis.
In einem ganzzahligen Ausdruck könnte die Verwendung der Sinus-Funktion demnach so aussehen:

ROUND(SIN(x))+y+z

Dabei sind die Variablen x, y und z vom Datentyp Integer; ebenso das Ergebnis von

ROUND(SIN(X))

Wie man leicht erkennt, muß man bei dieser Angelegenheit 'höllisch' aufpassen und genau wissen, was man mit seinen Daten anstellen will!
Mit dem jetzigen Wissen können wir Daten schon 'ganz schön' manipulieren (oder durcheinander bringen?). Aber irgendwie muß der Ablauf dieser Operationen in den meisten Fällen von Bedingungen abhängig gesteuert werden, die sich während der Programmausführung einstellen. Bedingungen – das bedeutet Vergleiche. Und Vergleichen bedeuten Boolesche Algebra und

Boolesche Ausdrücke

und wahr oder falsch.
Für die Konstruktion Boolescher Ausdrücke stehen uns die bis eben behandelten arithmetischen Ausdrücke, die Vergleichsoperatoren =, <,>, <=, >=, <> und die logischen Operatoren AND, OR und NOT zur Verfügung. Fangen wir klein an:
In der Booleschen Algebra wird nur mit den zwei Wahrheitswerten wahr (TRUE) und falsch (FALSE) gerechnet (s. Teil 3). Schauen wir uns ein paar einfache Boolesche Ausdrücke an:

a<>b

Dies ist ein Ausdruck mit einem Vergleichsoperator und liefert den Wert TRUE, wenn a und b unterschiedliche Werte haben. Dabei können die zwei Variablen vom Typ BOOLEAN oder von irgendeinem anderen Datentyp sein; als Ergebnis erhält man immer den Wert TRUE oder FALSE vom Typ BOOLEAN.

a OR b

Hier werden zwei Variablen vom Typ BOOLEAN mit dem Booleschen Operator OR (ODER) verknüpft. Als Ergebnis erhält man den Wert TRUE, wenn mindestens einer der zwei Operanten a oder b den Wert TRUE hat, sonst den Wert FALSE (s. auch Teil 3).

NOT a

Der Wahrheitswert der Booleschen Variablen a wird hier negiert. Hatte a vorher den Wert TRUE, so erhält die Variable den neuen Wert FALSE.

NOT a AND b

Hier wird zuerst a negiert und danach mit b undiert.

(a=0) OR (a=10)

Hier wird die numerische Variable a auf Gleichheit mit 0 oder 10 getestet. Entspricht sie einem der zwei Werte, so liefert der Ausdruck den Wert TRUE. Hier ist auch eine zwingende Vorschrift von Pascal bei der Bildung von Booleschen Ausdrücken zu erkennen:
Unterausdrücke müssen geklammert werden, sofern sie nicht mit NOT beginnen. Bsp.:

(x=y) AND (u+v=w) OR (c=d)

Durch die Klammerung von Booleschen Ausdrücken sind 'geschachtelte' Konstruktionen wie bei arithmetischen Ausdrücken möglich.
Deshalb ein paar Grundregeln der Booleschen Algebra:
NOT(NOT a)
ist äquivalent mit a.
NOT(a OR b)
mit (NOT a) AND (NOT b)
NOT(a AND b)
mit (NOT a) OR (NOT b)
Jetzt haben wir verhältnismäßig viele Operanten, die wir in Ausdrücken benutzen können (und werden), wie ist aber die Reihenfolge bzw. Priorität derselbigen bei der Auswertung durch Pascal? Hier die Hierarchie:

- Die Klammern stellen den Adel dar und lassen sich von keinem anderen Operator übertrumpfen.
- Das Boolesche NOT folgt in der Rangordnung auf dem Fuße.
- *, /, DIV, MOD und AND bilden den Mittelstand.
- Die Operatoren +, - und OR haben auch noch Privilegien.
- Schließlich bilden die Vergleichsoperatoren =, <, >, <=, >=, und <> das 'gemeine' Volk und müssen alles über sich ergehen lassen.

Bei der Bildung von komplexen Ausdrücken sind diese Prioritäten von immenser Bedeutung; allzuoft entstehen dabei Fehler. Ein Tip: Droht der Überblick wegen zuvieler Klammerebenen in einem Ausdruck verlorenzugehen, so sollte – ähnlich wie bei der schrittweisen Auflösung von mathematischen Gleichungssystemen, der Ausdruck in kleinere, überschaubarere Teilausdrücke zerlegt werden und sich über mehrere Programmanweisungen (kommt gleich!) erstrecken.
Resümieren wir noch einmal:
Ein Ausdruck kann eine einfache Konstante oder Variable sein:

5 a -1 +b

Nicht möglich ist das unmittelbare aufeinanderfolgen zweier Operatoren in einem Ausdruck wie bei:

4*-8

(4 multipliziert mit -8)
Hier müssen Klammern trennend einschreiten:

4*(-8)

Schließlich kann anstelle jeder Konstanten oder Variablen ein Funktionsaufruf eingesetzt werden, z.B.:

2*SIN(pi)+x

wobei, wie gesagt, auf die Verträglichkeit der Datentypen geachtet werden muß!
Nun können wir uns 'ausdrücken'; dem Rechner also hinsichtlich der Verknüpfung von Daten unsere Wünsche beibringen. Und weiter?

Anweisungen

werden erteilt!
In Pascal gibt es viele verschiedene Anweisungsarten. Von diesen haben wir in den vergangenen zwei Beispielprogrammen schon ein paar kennengelernt. So zum Beispiel die Anweisung:

READLN(wert);

Damit wurde eine Eingabe über die Standard-Eingabeeinheit Tastatur getätigt. Es ist also eine Eingabeanweisung. Das Gegenstück ist die Ausgabeanweisung:

WRITELN('Hallo');

die die Zeichenkettenkonstante Hallo auf das Standardausgabegerät gibt und die nächste Schreibposition auf eine neue Zeile legt. Mit:

WRITELN(2*a+b);

würde das Ergebnis des Ausdrucks:

2*a+b

ausgegeben. Wiederum zu einer anderen Art gehören folgende Anweisungen:

summe:=0;
ergebnis:=zahl1 + zahl2;

Bei diesen handelt es sich um Zuweisungen, mit der einer Variablen ein bestimmter Wert zugewiesen wird. Dazu wird der Ausdruck rechts vom Zuweisungsoperator := ausgewertet und der ermittelte Wert der Variablen links vom Zuweisungsope-

rator zugewiesen (gespeichert). Wie in der Einführung schon erwähnt, und von Programmieranfängern in BASIC auf Grund des Gleichheitszeichens als Zuweisungsoperator oft mißverstanden, hat eine Zuweisung mit der mathematischen Gleichsetzung nichts zu tun. Wegen der zeitlichen Abfolge des Geschehens sind Zuweisungen folgender Art möglich und sehr oft anzutreffen:

i:=i+x;

Hier wird zuerst der Ausdruck:

i+x

ausgewertet, wobei der aktuelle Wert der Variablen i (z.B. i=1) benutzt wird. Erst wenn der Ausdruck fertig abgearbeitet worden ist, wird der daraus resultierente neue Wert der Variablen i zugewiesen! Wenn also x den Wert 2 und i den Wert 1 vor Ausführung der Zuweisung besaßen, so enthält die Variable i danach den Wert 3 (1+2), während der Wert der Variablen x unverändert bleibt. Die Variable i behält nun den neuen Wert solange sie existiert oder einen neuen Wert zugewiesen bekommt.

Auch wenn es zunächst unsinnig erscheint – es gibt in Pascal auch eine leere Anweisung, in der rein, gar nichts passiert. Diese besteht nur aus dem Anweisungen trennenden Semikolon und sonst keinen weiteren Symbolen. Bsp:

monat:=10;;

1. Anweisung: monat:=10;
2. Anweisung: ;

Die leere Anweisung wird vom Compiler einfach überlesen, so daß man auch überflüssige Semikolons im Programm haben kann, wie im folgenden Fall:

```
monat:=10;
END.
```

Vor dem END braucht kein Semikolon zu stehen, da END (ebenso BEGIN) die gleiche Funktion wie das Semikolon haben: sie dienen als Trenn-Symbole von Anweisungen und stellen keine solchen dar.

Block

Bei Pascal handelt es sich um eine strukurierte Programmiersprache; ein Programm setzt sich aus verschiedenen, mehr oder weniger oft vorkommenden Programmstrukturen zusammen. Von diesen Strukturen haben wir auch schon die einfachste kennengelernt: die Befehlsfolge bzw. den Anweisungsblock.

Ein solcher Block wird mit den Symbolen BEGIN und END geklammert, z.B. so:

```
BEGIN
  WRITELN ('Wie geht es Dir ?');
  READLN (antwort);
END;
```

Ein solcher Block kann in einem Pascal-Programm überall dort eingesetzt werden, wo eine einzelne Anweisung stehen kann. Und ohne diese doch recht triviale Programmstruktur wären Pascalprogramme nicht möglich, wie sich zeigen wird.

Innerhalb des Blockes werden die enthaltenen Anweisungen, deren Anzahl theoretisch unbegrenzt ist, der Reihe nach (sequentiell) abgearbeitet; beginnend bei der Anweisung nach dem BEGIN bis zu der vor dem END. Wer aufgepaßt hat, wird sofort die funtamentale Eigenschaft einer strukturierten Sprache wie Pascal erkennen: Für einzelne Anweisungen können Anweisungsblöcke eingesetzt werden, in denen wiederum durch Blöcke ersetzbare Anweisungen enthalten sind. Damit ist die Möglichkeit gegeben, die zur Lösung eines Problems notwendigen Schritte logisch zu gliedern und zusammenzufassen...

(MC)

Basic-Kurs

Datensicherung ist die Aufgabe, die wir uns für den Basic-Kurs dieses Monats stellen. Nachdem unsere Adressverwaltung jetzt schon fast funktionierte, wollen wir in diesem Teil endlich einmal daran gehen, unsere mühsam eingetippten Daten für längere Zeiträume auf Kassette oder Diskette zu bannen.

Datensicherung erfolgt mit SAVE, haben wir in einem der ersten Teile gelernt. Dies trifft allerdings nur auf Programme zu. Variablen werden mit einer völlig anderen Befehlssequenz gespeichert.
Dazu müssen wir erst einmal etwas über die Kanäle (STREAMS) des Schneider erfahren. Insgesamt neun dieser Steuerkanäle stehen zur Verfügung. Auf sie können Input- und Print-Anweisungen verbogen werden. Dabei steht Stream 0 – 7 für Tastatur, Monitor und Windows zur Verfügung, der Kanal 8 für den Drucker und Kanal 9 für Datacorder bzw. Floppy.
Eine Ansammlung von Daten auf Diskette oder Kassette nennt man Datei. Um nun unsere Datei auf externe Datenträger zu bringen, müssen wir ihm angeben, daß die Datei eröffnet werden soll, danach mittels PRINT-Befehl die Daten in die Datei schreiben und sie dann wieder ordnungsgemäß schließen.
Zum Printen der Daten benutzen wir den Floppy-Stream Nr. 9. Die anderen Streams sind momentan noch uninteressant, wir kommen im nächsten Teil auf den Druckerstream Nr. 8 zurück.
Eine Print-Anweisung für einen Stream wird durch das deutsche Nummernzeichen # gekennzeichnet. Findet der Interpreter hinter einer Print-Anweisung (oder Input) ein solches Gatter, so weiß er, daß diese Ausgabe nicht auf den Bildschirm geleitet werden muß, sondern an ein Peripheriegerät.
Geöffnet wird eine Datei zur Ausgabe mit dem Befehl OPENOUT "DATEINAME". Unser Dateiname soll die Bezeichnung ADRESS.DAT tragen. Vorher soll aber noch der Bildschirm gelöscht und eine Meldung ausgegeben werden, daß der Computer mit dem Sichern der Daten beschäftigt ist.
Wir beginnen bei Zeile 3000 und schreiben:

3000 CLS
3010 LOCATE 10,10
3020 PRINT "DATEN SICHERN"

Danach wird die Datei zur Ausgabe eröffnet:
3030 OPENOUT "ADRESS.DAT"
Damit der Rechner beim Wiedereinladen noch weiß, wieviel Daten überhaupt gespeichert wurden, schreiben wir die laufende Nummer an den Anfang der Datei:
3040 PRINT#9,LFD
Die Variable LFD enthält bekanntlich die Anzahl der bisher erfaßten Daten und steht nun am Anfang unserer Datei.
Danach müssen wir zwei Schleifen ineinander verschachteln. Die erste Schleife muß hochzählen bis LFD und lautet:
3050 FOR FELD = 0 TO LFD
Die zweite Schleife muß innerhalb der ersten Schleife achtmal durchlaufen werden, und die Einträge des Feldes auf Datenträger schreiben:
3060 FOR ADRESS = 1 TO 8
3070 PRINT#9,ADR$(FELD, ADRESS)
3080 NEXT ADRESS
Danach durchläuft wieder Schleife 1:
3090 NEXT FELD
und schon kann die Datei wieder geschlossen werden
3100 CLOSEOUT
und zurück zum MENUE:
3110 RETURN

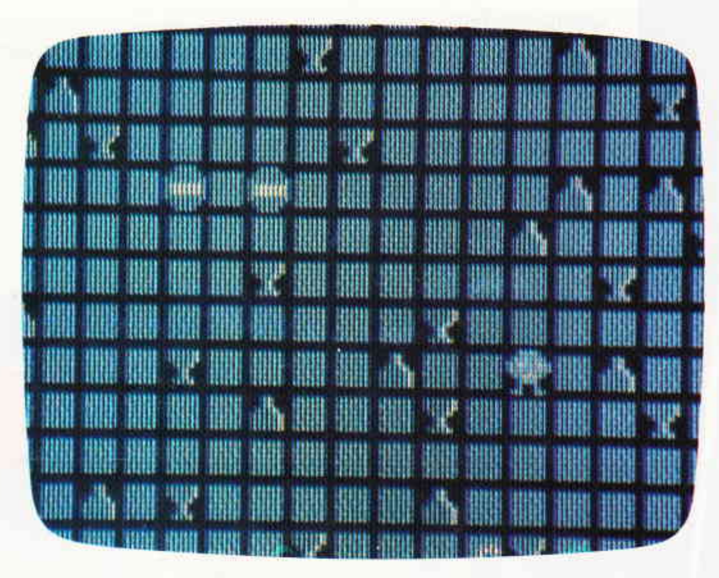
Das war schon alles zum Thema Datensicherung. Auf Ihrer Diskette/Kassette befindet sich jetzt eine Datei mit Namen ADRESS.DAT, die die Anzahl der bisher erfaßten Daten sowie das komplette Variablenfeld enthält.
Mit LOAD oder RUN können Sie sich diese Datei nicht mehr ansehen, sondern nur noch mit den Befehlen OPENIN, INPUT #9, CLOSEIN.
Ab Zeile 4000 steht in Listing 1 die Laderoutine für unseren Datenblock. Wie Sie sehen, hat sich außer den obengenannten Befehlen nichts gegenüber der Speicherroutine geändert. Lediglich INPUT wurde durch LINE INPUT vervollständigt, damit auch Variablen, die ein Komma enthalten, gelesen werden können.
Beim Abspeichern können Wartezeiten von ein paar Sekunden auftreten. Floppybesitzer, die den Namen ihrer Datei ändern wollen, sollten darauf achten, daß er nicht länger als acht Buchstaben wird.
Allgemein sollten jetzt alle Leser soweit sein, daß sie die INKEY-Routine aus den Tips und Tricks 7/86 bei Zeile 1510 einfügen können. Probieren Sie es mal aus. Wie es gemacht wird, erfahren Sie in der nächsten Lektion. (TM)

```
3000 CLS
3010 LOCATE 10,10
3020 PRINT"DATEN SICHERN"
3030 OPENOUT "adress.dat"
3040 PRINT#9,lfd
3050 FOR feld = 0 TO lfd
3060 FOR adress = 1 TO 8
3070 PRINT#9,adr$(feld,adress)
3080 NEXT adress
3090 NEXT feld
3100 CLOSEOUT
3110 RETURN
4000 CLS
4010 LOCATE 10,10
4020 PRINT"DATEN LADEN  "
4030 OPENIN "adress.dat"
4040 INPUT#9,lfd
4050 FOR feld = 0 TO lfd
4060 FOR adress = 1 TO 8
4070 LINE INPUT#9,adr$(feld,adress)
4080 NEXT adress
4090 NEXT feld
4100 CLOSEIN
4110 RETURN
```


SPIELE ZU GEWINNEN
Unter den Einsendern dieses Coupons werden 25 Exemplare des klassischen Arcade-Spiels AIRWOLF verlost.
Der Rechtsweg ist ausgeschlossen.
Name und Adresse:
Computer-Typ:
Name des Magazins, welchem Sie diesen Coupon entnommen haben:
Senden Sie diesen Coupon an:
PETER WEST RECORDS GmbH
Am Heerdter Hof 15
4000 Düsseldorf 11
GHOSTS'N GOBLINS
elite
SPACE INVASION
AMSTRAD/SCHNEIDER
elite
BOMB JACK
elite
BASIC
CONSTRUCTION
ARCADE
Der Knaller!!!!
Eine Basicerweiterung für den CPC 464 ohne RSX.
ACB erlaubt das einfache Programmieren von Arcade-Spielen.
Sehr einfach zu handhaben.
Alle Befehle werden direkt eingegeben.
Zu dem Paket gehören ein SPRITE EDITOR und ein
AUSFÜHRLICHES DEUTSCHES HANDBUCH.
Der Preis nur
44,90
Händleranfragen an:
PETER WEST RECORDS GmbH
Am Heerdter Hof 15
4000 Düsseldorf 11
Telefon 02 11/50 21 31
Telex 8 582 493 pwr oder
Telefax 02 11/5 04 86 19

Ortwins Run

"Ortwins Run" ist ein sehr schnelles Actionspiel mit fantastischer Grafik und vielen Features.
Helfen Sie Ortwin, seine Junggesellenbude in Ordnung zu bringen, bevor seine Freundin zu Besuch kommt.
Programmbeschreibung:
Wenn nach dem Intro das Menue erscheint, können Sie



sich zwischen einem fertigen Spiel (1) oder dem Games Designer (2) entscheiden. Haben Sie Punkt 1 angewählt, so erscheint ein Spielfeld, auf dem Gläser und Flaschen von Ortwins letzter Fete verstreut sind. Sammeln Sie diese Sachen ein, aber vermeiden Sie die Berührung mit Stopschildern und dem Spielfeldrand. Hier gibt es Energieabzug.
Für jedes eingesammelte Glas erhalten Sie 10 Punkte, für jede Flasche 50 Punkte. Haben Sie den ganzen Müll aufgeräumt, so erhalten Sie einen Zeitbonus. Am rechten Bildrand sind in einem Info-Window laufend die Punktzahl, der momentane Level und die verbleibende Restzeit sichtbar.
Haben Sie sich für Punkt 2 entschieden, so können Sie Ihr Spiel selber gestalten und erhalten ein neues Menue und das Spielfeld. Die sechs Menuepunkte des Designermenues haben folgende Funktion:

1. DESIGN:
Beim Anwählen dieses Menuepunkts wird abgefragt, mit welchem Objekt Sie arbeiten wollen. Fahren Sie dazu mit dem Fadenkreuz über das gewünschte Objekt und betätigen Sie den Feuerknopf. Wollen Sie eine neues Objekt anwählen, so betätigen Sie die Feuertaste. Punkt 7 bewirkt eine Rückkehr ins Hauptmenue.

2. STORE:
Hier wird die gewünschte Levelnummer angegeben und danach in den Speicher geschrieben. Dieser Vorgang dauert ca. 15 Sekunden.

3. SAVE:
Dieser Punkt dient zum Abspeichern des gerade konstruierten Spiels auf Datenträger.

4. LOAD:
Bei Anwahl dieses Menuepunktes können Sie Ihre Eigenkonstruktionen wieder laden.

5. PLAY:
Mit diesem Punkt können die Level im einzelnen durchgespielt werden. Sie geben einfach die Levelnummer an. Wurde das gewählte Level vorher nicht mit Store generiert, so erscheint eine Fehlermeldung.

6. BACK:
Dieser Menuepunkt bewirkt eine Rückkehr zum INTRO.

Steuerung:
Die Steuerung von Ortwin und dem Games Designer erfolgt entweder über Joystick oder die Cursortasten.
Floppybetrieb:
In den Zeilen 4190 und 4310 muß beim Openout- bzw. Openin-Befehl das Ausrufezeichen gegen einen für die Floppy gültigen Dateinamen ausgetauscht werden (max. 8 Zeichen bspw:"OGAME"). (I. Schwarze)

für 464-664-6128

```
10  ' ==================================        [2219]
20  ' =                                =        [286]
30  ' =    O R T W I N ' S    R U N    =        [764]
40  ' =                                =        [286]
50  ' = (c) 1986 northern software ltd =        [2456]
60  ' =                                =        [286]
70  ' =           written by           =        [524]
80  ' =                                =        [286]
90  ' =    JOCHEN 'JOHNNY' SCHWARZE    =        [1650]
100 ' =                                 =       [286]
110 ' ===================================       [2219]
120 '464 und 6128 BESITZER MUESSEN DIE VAR      [3461]
IABLE CURSOR$ IN KURSO$ AENDERN
130 ' ========= INIT ============               [1234]
140 '                                           [117]
150 SPEED WRITE 1                               [1302]
160 level=1:en=300:sc=0:hi=11111:it=0           [1632]
170 ortwin$="THE GREAT ORTWIN"                  [1574]
180 MODE 1                                      [506]
190 SYMBOL AFTER 32                             [1296]
200 MEMORY 34000                                [313]
210 INK 1,26                                    [56]
220 INK 2,11                                    [179]
230 INK 3,14                                    [228]
240 INK 0,0                                     [53]
250 PAPER 0:CLS                                 [973]
260 BORDER 0                                    [1008]
270 DIM feld(25,25)                             [539]
280 ENV 1,20,-2,10                              [734]
290 ENT 1,10,-2,1                               [588]
300 ENT 5,10,-2,14                              [588]
310 ENV 5,3,4,2,1,0,20,10,-0.5,10               [1512]
320 saml(1)=10:flas(1)=5                        [1046]
330 saml(2)=1:flas(2)=9                         [324]
340 saml(3)=15:flas(3)=15                       [1407]
350 saml(4)=1:flas(4)=1                         [983]
360 saml(5)=30:flas(5)=20                       [1417]
370 saml(6)=5:flas(6)=10                        [1083]
380 saml(7)=10:flas(7)=10                       [1702]
390 saml(8)=7:flas(8)=3                         [909]
400 saml(9)=40:flas(9)=10                       [1203]
410 hind(1)=5:hind(2)=10                        [1265]
420 hind(3)=20:hind(4)=100                      [1372]
430 hind(5)=5:hind(6)=60                        [822]
440 hind(7)=30:hind(8)=50                       [990]
450 hind(9)=80                                  [385]
460 '                                           [117]
470 ' ======= spielfigur =======                [2221]
480 '                                           [117]
490 SYMBOL 129,60,126,219,255,102,60,36,10      [2119]
2
500 SYMBOL 130,0,0,36,0,24                      [1359]
510 '                                           [117]
520 ' ======= hindernis ========                [1917]
530 '                                           [117]
540 SYMBOL 131,60,126,255,129,129,255,126,      [2078]
60
550 SYMBOL 132,0,0,0,126,126                    [1495]
560 '                                           [117]
570 '========= pokal ===========                [1699]
580 '                                           [117]
590 SYMBOL 134,0,126,126,60,24,24,60            [1882]
600 SYMBOL 135,0,2,2,4,8,8,4                    [2220]
610 '                                           [117]
620 '========= flasche =========                [1307]
630 '                                           [117]
640 SYMBOL 140,0,24,24,60,126,126,126           [2173]
650 SYMBOL 143,0,8,8,4,2,2,2                    [1556]
660 '                                           [117]
670 ' =========== erde =========                [1719]
680 '                                           [117]
690 SYMBOL 136,0,42,84,42,84,42,84              [2308]
700 SYMBOL 137,0,84,42,84,42,84,42              [2048]
```

Abo-Order
»ZEITSCHRIFT«
»DATABOX«

Abgesandt am:

________________ 198___

Bemerkungen:

____

____

____

____



Abo-Order
»ZEITSCHRIFT«
»DATABOX«

Das kompetente Magazin für alle Schneider-Anwender!

Bestellen Sie noch heute Ihr Abonnement mit dieser Postkarte!

Bitte ausreichend frankieren

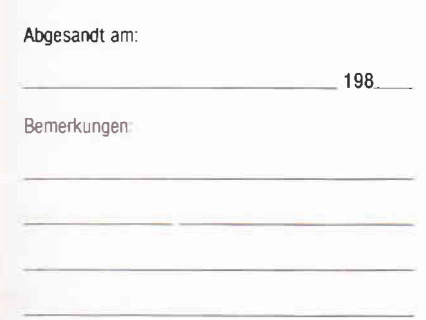
Antwortkarte

Schneider CPC International
Postfach 250

3440 Eschwege



»RESTPOSTEN-PACK 1985«
»Einzelhefte 1986«



Abgesandt am:

________________ 198___

Bemerkungen:

____

____

____

____



SONDERAKTION
»Restposten-Pack 1985«
»Einzelheftbestellung 1986«

Absender: *(Bitte genaue Anschrift angeben!)*

____
Name

____
Vorname

____
Firma

____
Straße/Nr./Postfach

____
PLZ/Ort



Bitte ausreichend frankieren

Antwortkarte

Schneider CPC International
Postfach 250

3440 Eschwege

Einzelbezug
»DATABOX«

Abgesandt am:

________________ 198___

Bemerkungen:

____

____

____

____




Einzelbezug
»DATABOX«

Absender: *(Bitte genaue Anschrift angeben!)*

____
Name

____
Vorname

____
Firma

____
Straße/Nr./Postfach

____
PLZ/Ort

Bitte ausreichend frankieren

Antwortkarte

Schneider CPC International
Postfach 250

3440 Eschwege

»Abo-Order Zeitschrift«

Hiermit bestelle ich „Schneider CPC International" für mindestens
☐ 1/2 Jahr ☐ 1 Jahr

Zustellung, Vertriebskosten und MwSt. sind im günstigen Preis von 30,- DM bzw. 60,- DM enthalten (BRD und West-Berlin).
Auslandspreise: **Europa** 12 Ausgaben – 90,- DM, 6 Ausgaben – 45,- DM
Außereuropäisches Ausland 12 Ausgaben – 120,- DM, 6 Ausgaben – 60,- DM
Lieferung soll ab Ausgabe Nr. .... erfolgen.

Lieferanschrift
☐ Geschäft ☐ Privat

Vorname, Name

Firma (nur wenn Lieferadresse)

Straße, Nr./Postfach

PLZ/Ort *(Bitte genaue Anschrift angeben!)*

Dieses Abonnement verlängert sich automatisch um 1/2 bzw. 1 Jahr, wenn es nicht sechs Wochen vor Ablauf schriftlich gekündigt wird.

Datum Unterschrift
(bei Minderjährigen: Unterschrift des ges. Vertreters)

Bitte unbedingt zwei Unterschriften leisten!

»Abo-Order DATABOX«

Hiermit bestelle ich die ☐ CPC Databox ☐ Joyce-Databox
für mindestens ☐ 1/2 Jahr ☐ 1 Jahr
als ☐ Cassette ☐ Diskette.

Bezugspreise Inland (einschl. Porto/Verpackung)
Cass. 1/2 Jahr 90,- DM, 1 Jahr 180,- DM
Disk. 3" 1/2 Jahr 150,- DM, 1 Jahr 300,- DM
Bezugspreise Ausland (eingeklammerte Werte außereuropäisch)
Cass. 1/2 Jahr 100,- (120,-) DM, 1 Jahr 200,- (240,-) DM
Disk. 3" 1/2 Jahr 160,- (180,-) DM, 1 Jahr 320,- (360,-) DM

Gewünschte Zahlungsweise:
☐ **Bequem und bargeldlos durch Bankabbuchung**

Bankleitzahl (von Scheck abschreiben)

Konto-Nr./-Inhaber

Geldinstitut

☐ **Gegen Rechnung** – zahlbar innerhalb zwei Wochen nach Erhalt.
(Bitte keine Vorauszahlung leisten – Rechnung abwarten.)

Garantie:
Ich weiß, daß ich diese Vereinbarung innerhalb einer Woche bei der Bestelladresse schriftlich widerrufen kann, wobei bereits die rechtzeitige Absendung meines Widerrufschreibens zur Fristwahrung ausreicht. Ich bestätige das durch meine zweite Unterschrift.

Datum Unterschrift

Abo-Zeitschrift
Abo-DATABOX

Bestellen Sie Ihr Abonnement mit der nebenstehenden Karte! Preisvorteil gegenüber dem Einzelbezug!
Mit einem Databox-Abo erhalten Sie die Software zum Magazin immer pünktlich und mit einem saftigen Preisvorteil gegenüber dem Einzelbezug!

Sonderaktion Restposten-Pack 1985

☐ Aus Restbeständen des Jahrgangs 1985 der Zeitschrift „Schneider CPC International" bestelle ich hiermit die noch vorhandenen vier Ausgaben
8/85 10/85 11/85 12/85
im Paket zum günstigen Gesamtpreis von nur 12,– DM (Ausland 15,– DM). In diesem Preis sind die Kosten für Porto und Verpackung bereits enthalten. Lieferung ist nur per Vorkasse möglich (V-Scheck bitte beifügen).
Nur solange der Vorrat reicht!

Jahrgang 1986

Die Sonderhefte sowie sämtliche Ausgaben von „Schneider CPC International" des Jahrgangs 1986 sind über den Verlag zu beziehen. Bei einem Bestellwert von mindestens 15,– DM werden keine Porto- und Versandgebühren erhoben; bei einem Bestellwert unter 15,– DM werden 3,– DM Porto/Verpackung berechnet (Ausland 5,– DM Porto/Verpackung). Lieferung nur gegen Vorkasse (V-Scheck).
Bestellung 1986'er Ausgaben (bitte ankreuzen):

☐ 1/86	5,– DM
☐ 2/86	5,– DM
☐ 3/86	5,– DM
☐ 4/86	5,– DM
☐ 5/86	6,– DM
☐ 6/86	6,– DM
☐ 7/86	6,– DM
☐ Sonderheft 1/86	14,– DM
☐ Sonderheft 2/86	14,– DM
Porto/Verpackung (nur bei einem Bestellwert unter 15,– DM)	DM
Gesamtbetrag	DM

☐ Einen Verrechnungsscheck in Höhe des Rechnungsbetrages habe ich beigefügt.

Datum: Unterschrift (bei Minderjährigen des ges. Vertreters)

Bestellkarte
»RESTPOSTEN-PACK 1985«
»EINZELHEFTE 1986«

Wir räumen unser Lager und bieten unseren Lesern, die 1985 noch nicht "dabei" waren, die Möglichkeit, die noch vorhandenen Ausgaben des vergangenen Jahres preisgünstig "im Paket" zur erstehen. Dies sind die Ausgaben 8, 10, 11 und 12/85.
Bitte benutzen Sie hierfür die nebenstehende Karte!
Die 1986'er Ausgaben sind noch sämtlich erhältlich.

»Databox & Databox Highlights«
Einzelbestellung

Ich bestelle hiermit die Schneider Databox (bitte ankreuzen)

CPC	Kassette	Diskette 3"
Ausgabe 8/85	☐ 14,– DM	
Ausgabe 9/85	☐ 14,– DM	
Ausgabe 10/85	☐ 14,– DM	
Ausgabe 11/85	☐ 14,– DM	
Ausgabe 12/85	☐ 14,– DM	
Ausgabe 1/86	☐ 14,– DM	
Ausgabe 2/86	☐ 14,– DM	2/86 ☐ 24,– DM
Ausgabe 3/86	☐ 14,– DM	3/86 ☐ 24,– DM
Ausgabe 4/86	☐ 14,– DM	4/86 ☐ 24,– DM
Ausgabe 5/86	☐ 14,– DM	5/86 ☐ 24,– DM
Ausgabe 6/86	☐ 14,– DM	6/86 ☐ 24,– DM
Ausgabe 7/86	☐ 14,– DM	7/86 ☐ 24,– DM
Ausgabe 8/86	☐ 14,– DM	8/86 ☐ 24,– DM
Highlights 1	☐ 14,– DM	
Highlights 2	☐ 14,– DM	
NEU		
Joyce 5/86		5/86 ☐ 24,– DM
Joyce 6/86		6/86 ☐ 24,– DM
Joyce 7/86		7/86 ☐ 24,– DM
Joyce 8/86		8/86 ☐ 24,– DM
+ Porto/Verpackung (Porto/Verp. Inland 3,– DM) (Porto/Verp. Ausland 5,– DM)		DM
Gesamtbetrag		DM

☐ Diesen Betrag bezahle ich mittels des beigefügten Verrechnungsschecks.
☐ Ich bitte um Lieferung per Nachnahme (nur im Inland).
Bei Nachnahme kommt zum o.g. Betrag noch die Nachnahmegebühr hinzu.

Datum Unterschrift (bei Minderjährigen des gesetzl. Vertreters)

DATABOX – jetzt auch im Abonnement erhältlich.
Bitte Bestellkarte benutzen!

»DATABOX«
Einzelbezug

Die Software zum Magazin!
Problemlos bestellen mit der Anforderungskarte!

```
710 '                                          [117]
720 ' ========== rand ==========               [1748]
730 '                                          [117]
740 SYMBOL 133,255,129,129,129,129,129,129     [2674]
,255
750 '                                          [117]
760 ' =====spielfigur/shape=====               [1940]
770 '                                          [117]
780 mann$=CHR$(32)+CHR$(8)+CHR$(22)+CHR$(1     [6642]
)+CHR$(15)+CHR$(3)+CHR$(129)+CHR$(8)+CHR$(
15)+CHR$(1)+CHR$(130)+CHR$(22)+CHR$(0)
790 '                                          [117]
800 ' =====hindernis/shape======               [1896]
810 '                                          [117]
820 shape$(1)=CHR$(32)+CHR$(8)+CHR$(22)+CH     [5352]
R$(1)+CHR$(15)+CHR$(2)+CHR$(131)+CHR$(8)+C
HR$(15)+CHR$(1)+CHR$(132)+CHR$(22)+CHR$(0)

830 '                                          [117]
840 ' ========pokal/shape=======               [2348]
850 '                                          [117]
860 shape$(2)=CHR$(32)+CHR$(8)+CHR$(22)+CH     [7882]
R$(1)+CHR$(15)+CHR$(2)+CHR$(134)+CHR$(8)+C
HR$(15)+CHR$(1)+CHR$(135)+CHR$(22)+CHR$(0)
870 '                                          [117]
880 ' ========flasche/shape=====               [2228]
890 '                                          [117]
900 shape$(4)=CHR$(32)+CHR$(8)+CHR$(22)+CH     [6900]
R$(1)+CHR$(15)+CHR$(2)+CHR$(140)+CHR$(8)+C
HR$(15)+CHR$(1)+CHR$(143)+CHR$(22)+CHR$(0)
910 '                                          [117]
920 ' =========erde/shape=======               [1973]
930 '                                          [117]
940 shape$(0)=CHR$(32)+CHR$(8)+CHR$(22)+CH     [7128]
R$(1)+CHR$(15)+CHR$(3)+CHR$(136)+CHR$(8)+C
HR$(15)+CHR$(2)+CHR$(137)+CHR$(22)+CHR$(0)
950 '                                          [117]
960 ' =========rand/shape=======               [1797]
970 '                                          [117]
980 shape$(3)=CHR$(32)+CHR$(8)+CHR$(22)+CH     [8082]
R$(1)+CHR$(15)+CHR$(3)+CHR$(136)+CHR$(8)+C
HR$(15)+CHR$(2)+CHR$(137)+CHR$(8)+CHR$(15)

+CHR$(1)+CHR$(133)+CHR$(22)+CHR$(0)
990 items(1)=15:items(2)=10                    [2003]
1000 items(3)=30:items(4)=2                    [1029]
1010 items(5)=50:items(6)=15                   [1716]
1020 items(7)=20:items(8)=10                   [1646]
1030 items(9)=50                               [569]
1040 sc=0:level=1:it=0                         [1251]
1050 '                                         [117]
1060 '===================== intro == ====      [3929]
================
1070 '                                         [117]
1080 MODE 1                                    [506]
1090 POKE &B28B,120                            [659]
1100 FOR a=1 TO 40:LOCATE a,1:PRINT shape$     [1575]
(0):NEXT
1110 FOR a=1 TO 25:LOCATE 40,a:PRINT shape     [1883]
$(0):NEXT
1120 FOR a=40 TO 1 STEP -1:LOCATE a,25:PRI     [3855]
NT shape$(0):NEXT
1130 FOR a=25 TO 1 STEP -1:LOCATE 1,a:PRIN     [3645]
T shape$(0):NEXT
1140 PEN 1                                     [549]
1150 PRINT CHR$(22);"1"                        [1293]
1160 FOR a=1 TO 40:LOCATE a,1:PRINT CHR$(1     [1669]
33):NEXT
1170 FOR a=1 TO 25:LOCATE 40,a:PRINT CHR$(     [2168]
133):NEXT
1180 FOR a=40 TO 1 STEP -1:LOCATE a,25:PRI     [4356]
NT CHR$(133):NEXT
1190 FOR a=25 TO 1 STEP -1:LOCATE 1,a:PRIN     [2881]
T CHR$(133):NEXT
1200 PRINT CHR$(22);"0"                        [1295]
1210 LOCATE 10,3:PRINT"NORTHERN SOFTWARE L     [3113]
TD."
1220 LOCATE 17,5:PRINT"PRESENTS"               [1676]
1230 LOCATE 15,8:PRINT"ORTWIN'S RUN"           [1813]
1240 LOCATE 16,12:PRINT"WRITTEN BY"            [1799]
1250 LOCATE 18,14:PRINT"JOHNNY"                [1694]
1260 LOCATE 18,19:PRINT CHR$(164);" 1986"      [1618]
1270 GOSUB 4960                                [923]
1280 LOCATE 11,23:PRINT"ANY KEY TO CONTINU     [2547]
E."
1290 CALL &BB06                                [393]
1300 '                                         [117]
1310 '=============== menue ==============     [2286]
==
1320 '                                         [117]
1330 WINDOW#1,2,39,2,24:CLS#1                  [1123]
1340 LOCATE 4,3:PRINT"PLAY COMPUTER-STYLED     [4052]
GAME ...: 1"
1350 LOCATE 4,5:PRINT"GAME DESIGNER ......     [2958]
.........: 2"
1360 LOCATE 28,23:PRINT"GO ON ..."             [1203]
1370 a$="":WHILE a$="":a$=INKEY$:WEND          [1812]
1380 a=VAL(a$):IF a<1 OR a>2 THEN 1370         [1353]
1390 IF a=1 THEN GOTO 1440                     [1703]
1400 IF a=2 THEN GOTO 3090                     [880]
1410 '                                         [117]
1420 ' ======= spielfeld =======               [1859]
1430 '                                         [117]
1440 MODE 1:selfstyled=0:level=1:en=300        [3301]
1450 '                                         [117]
1460 FOR y=1 TO 25:FOR x=1 TO 25:feld(x,y)     [4570]
=0:LOCATE x,y:PRINT shape$(0):NEXT x,y
1470 LOCATE 1,1:PRINT CHR$(22);"1";:PEN 1      [1097]
1480 FOR a=1 TO 25                             [931]
1490 LOCATE a,1:PRINT CHR$(133);:feld(a,1)     [1521]
=3
1500 LOCATE 25+1-a,25:PRINT CHR$(133);:fel     [3263]
d(25+1-a,25)=3
1510 LOCATE 25,a:PRINT CHR$(133);:feld(25,     [3250]
a)=3
1520 LOCATE 1,25+1-a:PRINT CHR$(133);:feld     [3303]
(1,25+1-a)=3
1530 NEXT:PRINT CHR$(22);"0"                   [1782]
1540 '                                         [117]
1550 ' = hindernisse verteilen ==              [1475]
1560 '                                         [117]
1570 FOR i=1 TO hind(level)                    [2191]
1580 x=INT(RND*22)+2                           [1431]
1590 y=INT(RND*22)+2                           [1437]
1600 IF feld(x+1,y)<>0 THEN 1580               [1629]
1610 IF feld(x-1,y)<>0 THEN 1580               [906]
1620 IF feld(x,y+1)<>0 THEN 1580               [2148]
1630 IF feld(x,y-1)<>0 THEN 1580               [987]
1640 IF feld(x,y)<>0 THEN 1580                 [1644]
```



```
1650 feld(x,y)=1                                   [574]
1660 LOCATE x,y:PRINT shape$(1)                    [1467]
1670 NEXT i                                        [375]
1680 '                                             [117]
1690 ' ==== pokale verteilen ====                  [3338]
1700 '                                             [117]
1710 FOR i=1 TO saml(level)                        [982]
1720 x=INT(RND*22)+2                               [1431]
1730 y=INT(RND*22)+2                               [1437]
1740 IF feld(x+1,y)<>0 THEN 1720                   [1681]
1750 IF feld(x-1,y)<>0 THEN 1720                   [862]
1760 IF feld(x,y+1)<>0 THEN 1720                   [2088]
1770 IF feld(x,y-1)<>0 THEN 1720                   [847]
1780 IF feld(x,y)<>0 THEN 1720                     [1664]
1790 feld(x,y)=2                                   [573]
1800 LOCATE x,y:PRINT shape$(2)                    [1549]
1810 NEXT i                                        [375]
1820 '                                             [117]
1830 ' == flaschen verteilen ===                   [1941]
1840 '                                             [117]
1850 FOR i=1 TO flas(level)                        [588]
1860 x=INT(RND*22)+2                               [1431]
1870 y=INT(RND*22)+2                               [1437]
1880 IF feld(x+1,y)<>0 THEN 1860                   [1527]
1890 IF feld(x-1,y)<>0 THEN 1860                   [898]
1900 IF feld(x,y+1)<>0 THEN 1860                   [2140]
1910 IF feld(x,y-1)<>0 THEN 1860                   [1138]
1920 IF feld(x,y)<>0 THEN 1860                     [1636]
1930 feld(x,y)=4                                   [571]
1940 LOCATE x,y:PRINT shape$(4)                    [1545]
1950 NEXT i                                        [375]
1960 '                                             [117]
1970 ' ============= anzeige rechts ====== [2570]
=============
1980 '                                             [117]
1990 PEN 1                                         [549]
2000 LOCATE 27,10:PRINT"SCORE : ";USING"## [1394]
###";sc
2010 LOCATE 27,13:PRINT"HIGH  : ";USING"## [2360]
###";hi
2020 LOCATE 27,16:PRINT"LEVEL : ";USING"## [2707]
###";level
2030 LOCATE 27,19:PRINT"ITEMS : ";USING"## [3173]
###";it
2040 LOCATE 27,22:PRINT"ENERGY: ";USING"## [3113]
###";en
2050 '                                             [117]
2060 '================= HAUPTPROGRAMM ==== [3092]
=============
2070 '                                             [117]
2080 erde$=shape$(0)                               [1231]
2090 '                                             [117]
2100 '=========== spielfigur setzen ====== [3521]
=============
2110 '                                             [117]
2120 x=INT(RND*22)+2:y=INT(RND*22)+2               [1866]
2130 IF feld(x+1,y)<>0 THEN 2120                   [1555]
2140 IF feld(x-1,y)<>0 THEN 2120                   [862]
2150 IF feld(x,y+1)<>0 THEN 2120                   [2088]
2160 IF feld(x,y-1)<>0 THEN 2120                   [846]
2170 IF feld(x,y)<>0 THEN 2120                     [1664]
2180 '                                             [117]
2190 '===== joystick + cursortasten abfrag [2698]
en =====
2200 '                                             [117]
2210 beweg=0:LOCATE x,y:PRINT mann$                [823]
2220 IF en<1 THEN 2720                             [710]
2230 IF items(level)=0 THEN 2850                   [1059]
2240 IF JOY(0)=1 OR INKEY(0)=0 THEN beweg= [2738]
1:' oben
2250 IF JOY(0)=2 OR INKEY(2)=0 THEN beweg= [2606]
2:' unten
2260 IF JOY(0)=4 OR INKEY(8)=0 THEN beweg= [3067]
3:' links
2270 IF JOY(0)=8 OR INKEY(1)=0 THEN beweg= [2653]
4:' rechts
2280 '                                             [117]
2290 ' ========== bewegung nach oben ===== [2165]
2300 '                                             [117]
2310 IF beweg<>1 THEN 2410                         [524]
2320 SOUND 1,100,2,5 ,,,15                         [1550]
2330 IF feld(x,y-1)=0 THEN y=y-1:LOCATE x, [4182]
y:PRINT mann$:LOCATE x,y+1:PRINT erde$
2340 IF feld(x,y-1)=2 THEN feld(x,y-1)=0:S [15863]
OUND 1,40,10,15,,1:sc=sc+10:it=it+1:items(
level)=items(level)-1:y=y-1:LOCATE x,y:PRI
NT mann$:LOCATE x,y+1:PRINT erde$;:LOCATE
35,10:PEN 1:PRINT USING"#####";sc:LOCATE 3
5,19:PRINT USING"#####";it:GOTO 2220
2350 IF feld(x,y-1)=4 THEN feld(x,y-1)=0:S [12507]
OUND 1,40,10,15,,1:sc=sc+50:it=it+1:items(
level)=items(level)-1:y=y-1:LOCATE x,y:PRI
NT mann$:LOCATE x,y+1:PRINT erde$;:LOCATE
35,10:PEN 1:PRINT USING"#####";sc:LOCATE 3
5,19:PRINT USING"#####";it:GOTO 2220
2360 IF feld(x,y-1)=3 THEN en=en-1:LOCATE [4093]
35,22:PEN 1:PRINT USING"#####";en:GOTO 222
0
2370 IF feld(x,y-1)=1 THEN en=en-1:LOCATE [4050]
35,22:PEN 1:PRINT USING"#####";en:GOTO 222
0
2380 '                                             [117]
2390 ' ========== bewegung nach unten ==== [1860]
2400 '                                             [117]
2410 IF beweg<>2 THEN 2510                         [1564]
2420 SOUND 1,100,2, 5,,,15                         [1550]
2430 IF feld(x,y+1)=0 THEN y=y+1:LOCATE x, [6986]
y:PRINT mann$:LOCATE x,y-1:PRINT erde$
2440 IF feld(x,y+1)=2 THEN feld(x,y+1)=0:S [15243]
OUND 1,40,10,15,,1:sc=sc+10:it=it+1:items(
level)=items(level)-1:y=y+1:LOCATE x,y:PRI
NT mann$:LOCATE x,y-1:PRINT erde$;:LOCATE
35,10:PEN 1:PRINT USING"#####";sc:LOCATE 3
5,19:PRINT USING"#####";it:GOTO 2220
2450 IF feld(x,y+1)=4 THEN feld(x,y+1)=0:S [14180]
OUND 1,40,10,15,,1:sc=sc+50:it=it+1:items(
level)=items(level)-1:y=y+1:LOCATE x,y:PRI
NT mann$:LOCATE x,y-1:PRINT erde$;:LOCATE
35,10:PEN 1:PRINT USING"#####";sc:LOCATE 3
5,19:PRINT USING"#####";it:GOTO 2220
2460 IF feld(x,y+1)=3 THEN en=en-1:LOCATE [4862]
35,22:PEN 1:PRINT USING"#####";en:GOTO 222
0
2470 IF feld(x,y+1)=1 THEN en=en-1:LOCATE [6874]
35,22:PEN 1:PRINT USING"#####";en:GOTO 222
0
2480 '                                             [117]
2490 ' ========== bewegung nach links ==== [2165]
2500 '                                             [117]
2510 IF beweg<>3 THEN 2610                         [730]
2520 SOUND 1,100,2, 5,,,15                         [1550]
2530 IF feld(x-1,y)=0 THEN x=x-1:LOCATE x, [4571]
y:PRINT mann$:LOCATE x+1,y:PRINT erde$
2540 IF feld(x-1,y)=2 THEN feld(x-1,y)=0:S [13008]
OUND 1,40,10,15,,1:sc=sc+10:it=it+1:items(
level)=items(level)-1:x=x-1:LOCATE x,y:PRI
NT mann$:LOCATE x+1,y:PRINT erde$;:LOCATE
35,10:PEN 1:PRINT USING"#####";sc:LOCATE 3
5,19:PRINT USING"#####";it:GOTO 2220
2550 IF feld(x-1,y)=4 THEN feld(x-1,y)=0:S [13139]
OUND 1,40,10,15,,1:sc=sc+50:it=it+1:items(
level)=items(level)-1:x=x-1:LOCATE x,y:PRI
NT mann$:LOCATE x+1,y:PRINT erde$;:LOCATE
35,10:PEN 1:PRINT USING"#####";sc:LOCATE 3
5,19:PRINT USING"#####";it:GOTO 2220
2560 IF feld(x-1,y)=3 THEN en=en-1:LOCATE [3067]
35,22:PEN 1:PRINT USING"#####";en:GOTO 222
0
2570 IF feld(x-1,y)=1 THEN en=en-1:LOCATE [2348]
35,22:PEN 1:PRINT USING"#####";en:GOTO 222
0
2580 '                                             [117]
2590 ' ========== bewegung nach rechts === [1874]
2600 '                                             [117]
2610 IF beweg<>4 THEN 2680                         [692]
2620 SOUND 1,100,2, 5,,,15                         [1550]
2630 IF feld(x+1,y)=0 THEN x=x+1:LOCATE x, [6379]
y:PRINT mann$:LOCATE x-1,y:PRINT erde$
2640 IF feld(x+1,y)=2 THEN feld(x+1,y)=0:S [14653]
OUND 1,40,10,15,,1:sc=sc+10:it=it+1:items(
level)=items(level)-1:x=x+1:LOCATE x,y:PRI
NT mann$:LOCATE x-1,y:PRINT erde$;:LOCATE
35,10:PEN 1:PRINT USING"#####";sc:LOCATE 3
5,19:PRINT USING"#####";it:GOTO 2220
2650 IF feld(x+1,y)=4 THEN feld(x+1,y)=0:S [15329]
OUND 1,40,10,15,,1:sc=sc+50:it=it+1:items(
level)=items(level)-1:x=x+1:LOCATE x,y:PRI
NT mann$:LOCATE x-1,y:PRINT erde$;:LOCATE
35,10:PEN 1:PRINT USING"#####";sc:LOCATE 3
5,19:PRINT USING"#####";it:GOTO 2220
2660 IF feld(x+1,y)=3 THEN en=en-1:LOCATE [5830]
35,22:PEN 1:PRINT USING"#####";en:GOTO 222
0
2670 IF feld(x+1,y)=1 THEN en=en-1:LOCATE [5118]
35,22:PEN 1:PRINT USING"#####";en:GOTO 222
0
2680 '                                             [117]
2690 '=========== game over ==========         [2533]
```

```
2700 '                                       [117]
2710 GOTO 2220                               [367]
2720 LOCATE 6,11:PRINT STRING$(15,32)        [1725]
2730 LOCATE 6,12:PRINT STRING$(15,32)        [1579]
2740 LOCATE 6,13:PRINT STRING$(15,32)        [1338]
2750 PLOT 80,192,1:DRAW 318,192:DRAW 318,2   [4106]
38:DRAW 80,238:DRAW 80 192
2760 PLOT 82,194,2:DRAW 316,194:DRAW 316,2   [3817]
36:DRAW 82,236:DRAW 82,194
2770 PLOT 84,196,3:DRAW 314,196:DRAW 314,2   [3803]
34:DRAW 84,234:DRAW 84,196
2780 LOCATE 9,12:PEN 1:PRINT"GAME OVER"      [1464]
2790 IF sc>hi THEN hi=sc                     [1361]
2800 SOUND 1,600,140,15,5,5:SOUND 2,604,14   [3930]
0,15,5,5:SOUND 4,596,1 0,15,5,5
2810 FOR a=1 TO 2000:NEXT:IF selfstyled=1    [4402]
THEN 3090 ELSE 990
2820 '                                       [117]
2830 '==== level beendet =============       [2514]
2840 '                                       [117]
2850 LOCATE 6,10:PRINT STRING$(15,32)        [1609]
2860 LOCATE 6,11:PRINT STRING$(15,32)        [1725]
2870 LOCATE 6,12:PRINT STRING$(15,32)        [1579]
2880 LOCATE 6,13:PRINT STRING$(15,32)        [1338]
2890 LOCATE 6,14:PRINT STRING$(15,32)        [1822]
2900 LOCATE 6,15:PRINT STRING$(15,32)        [1769]
2910 LOCATE 6,16:PRINT STRING$(15,32)        [2176]
2920 PLOT 80,144,1:DRAW 318,144:DRAW 318,2   [3347]
54:DRAW 80,254:DRAW 80,144
2930 PLOT 82,146,2:DRAW 316,146:DRAW 316,2   [2894]
52:DRAW 82,252:DRAW 82,146
2940 PLOT 84,148,3:DRAW 314,148:DRAW 314,2   [4381]
50:DRAW 84,250:DRAW 84,148
2950 LOCATE 11,11:PEN 1:PRINT"BONUS"         [2172]
2960 zahl=level*10                           [1364]
2970 bonus=level*100                         [1358]
2980 LOCATE 7,13:PRINT zahl;"x ";shape$(2)   [5672]
;" =1";bonus:LOCATE 9,15:PRINT"+ ";:PRINT
 USING"###";en;:PRINT" x ";shape$(4)
2990 FOR i=1 TO 8:FOR a=26 TO 1 STEP -1:IN   [17526]
K 0,26-a:BORDER 26-a:SOUND 1,(a+20*3),1.4,
15:NEXT:NEXT:BORDER 0:INK 0,0:level=level+
1:sc=sc+bonus+en:LOCATE 35,10:PRINT USING"
#####";sc:IF level=10 THEN 4550 ELSE en=30
0:IF selfstyled=1 THEN FOR a=1 TO 3000:NEX
T
3000 IF selfstyled=1 THEN GOTO 3010 ELSE G   [2225]
OTO 1460
3010 '                                       [117]
3020 '                                       [117]
3030 '=================== hauptprogramm I   [5040]
I ==================================
3040 '                                       [117]
3050 '================== game - designer     [4016]
===================================
3060 '                                       [117]
3070 '================== menue ==========    [2109]
===================================
3080 '                                       [117]
3090 MODE 1                                  [506]
3100 FOR i=1 TO 25:FOR ii=1 TO 25            [1280]
3110 LOCATE ii,i:PRINT shape$(0);:feld(ii,   [2137]
i)=0
3120 NEXT ii,i                               [438]
3130 PEN 1:PRINT CHR$(22);"1":POKE &B28B,1   [3283]
20
3140 FOR a=1 TO 25                           [931]
3150 LOCATE a,1:PRINT CHR$(133);:feld(a,1)   [1521]
=3
3160 LOCATE 25,a:PRINT CHR$(133);:feld(25,   [3250]
a)=3
3170 LOCATE 26-a,25:PRINT CHR$(133);:feld(   [2550]
26-a,25)=3
3180 LOCATE 1,26-a:PRINT CHR$(133);:feld(1   [1972]
,26-a)=3
3190 NEXT a                                  [383]
3200 LOCATE 27,2:PRINT"GAME DESIGNER"        [1810]
3210 LOCATE 28,7:PRINT"DESIGN : 1"           [2364]
3220 LOCATE 28,9:PRINT"STORE  : 2"           [1759]
3230 LOCATE 28,11:PRINT"SAVE   : 3"          [1307]
3240 LOCATE 28,13:PRINT"LOAD   : 4"          [1298]
3250 LOCATE 28,15:PRINT"PLAY   : 5"          [1173]
3260 LOCATE 28,17:PRINT"BACK   : 6"          [1723]
3270 LOCATE 28,20:PRINT"GO ON ..."           [1526]
3280 a$=INKEY$:a=VAL(a$)                     [759]
3290 IF a<1 OR a>6 THEN 3280                 [922]
3300 ON a GOTO 3340,3840,4100,4230,4350,99   [1349]
0
3310 '                                       [117]
```

```
3320 '===================== designer =====  [3019]
===================
3330 '                                       [117]
3340 SYMBOL 255,16,16,0,214,0,16,16         [2213]
3350 cursor$=CHR$(255):shape$(5)=CHR$(22)+  [2853]
"0"+" "
3360 x=2:y=2                                 [803]
3370 WINDOW #1,27,40,1,25:CLS#1             [1386]
3380 LOCATE 30,12:PRINT shape$(1);" = 1"    [2607]
3390 LOCATE 30,14:PRINT shape$(2);" = 2"    [2214]
3400 LOCATE 30,16:PRINT shape$(4);" = 3"    [2477]
3410 LOCATE 30,18:PRINT shape$(0);:PEN 1:P  [2493]
RINT" = 4"
3420 LOCATE 30,20:PRINT shape$(3);" = 5"    [2709]
3430 PEN 1:LOCATE 30,22:PRINT"   = 6"       [1982]
3440 LOCATE 28,24:PRINT"END = 7"            [1310]
3450 PEN 1:LOCATE 27,2:PRINT"SELECT SHAPE"  [1964]
3460 LOCATE 27,4:PRINT"-->     ":shape$=INK [4928]
EY$:shape=VAL(shape$)
3470 IF shape<1 OR shape>7 THEN 3460        [1460]
3480 IF shape=7 THEN 3790                    [787]
3490 LOCATE 31,4:PRINT shape                 [854]
3500 IF shape=1 THEN sh=1                   [1085]
3510 IF shape=2 THEN sh=2                    [567]
3520 IF shape=3 THEN sh=4                   [1104]
3530 IF shape=4 THEN sh=0                   [1351]
3540 IF shape=5 THEN sh=3                   [1095]
3550 IF shape=6 THEN sh=5                    [316]
3560 LOCATE x,y:PEN 1:PRINT CHR$(22);"1";c  [1473]
ursor$;CHR$(22);"0"
3570 FOR verzoe=1 TO 20:NEXT                [1488]
3580 IF JOY(0)<>1 AND INKEY(0)<>0 THEN 362  [2441]
0
3590 y=y-1:IF y<1 THEN y=1                  [1834]
3600 LOCATE x,y+1:PRINT shape$(feld(x,y+1)  [1826]
)
3610 GOTO 3560                               [465]
3620 IF JOY(0)<>2 AND INKEY(2)<>0 THEN 366  [3017]
0
3630 y=y+1:IF y>25 THEN y=25                [1338]
3640 LOCATE x,y-1:PRINT shape$(feld(x,y-1)  [2953]
)
3650 GOTO 3560                               [465]
3660 IF JOY(0)<>4 AND INKEY(8)<>0 THEN 370  [2306]
0
3670 x=x-1:IF x<1 THEN x=1                  [1791]
3680 LOCATE x+1,y:PRINT shape$(feld(x+1,y)  [2156]
)
3690 GOTO 3560                               [465]
3700 IF JOY(0)<>8 AND INKEY(1)<>0 THEN 374  [3065]
0
3710 x=x+1:IF x>25 THEN x=25                [2727]
3720 LOCATE x-1,y:PRINT shape$(feld(x-1,y)  [2310]
)
3730 GOTO 3560                               [465]
3740 IF JOY(0)<>16 AND INKEY(9) THEN 3770   [1639]
3750 feld(x,y)=sh                           [1008]
3760 GOTO 3560                               [465]
3770 IF INKEY(47)=0 THEN GOTO 3460           [925]
3780 IF INKEY(41)<>0 THEN 3560              [1120]
3790 LOCATE x,y:PRINT shape$(feld(x,y))     [2370]
3800 CLS#1:PEN 1:GOTO 3200                   [885]
3810 '                                       [117]
3820 ' ========================= store ==   [3478]
=========================
3830 '                                       [117]
3840 WINDOW#1,26,40,1,25:CLS#1              [1909]
3850 LOCATE 28,2:PRINT"WHICH LEVEL"         [2305]
3860 LOCATE 28,4:PRINT"TO STORE ?"          [1845]
3870 LOCATE 28,6:INPUT"->",levelnr$         [1375]
3880 levelnr=VAL(levelnr$):IF levelnr<1 OR [4236]
 levelnr>9 THEN CLS#1:GOTO 3850
3890 pokal(levelnr)=0:flasche(levelnr)=0    [1577]
3900 LOCATE 28,9:PRINT"PLEASE WAIT."        [2570]
3910 FOR y=1 TO 25                          [1112]
3920 FOR x=1 TO 25                           [768]
3930 POKE (35000+(levelnr-1)*628)+((y-1)*2  [3043]
5)+x,feld(x,y)
3940 IF feld(x,y)=2 THEN pokal(levelnr)=po  [3692]
kal(levelnr)+1
3950 IF feld(x,y)=4 THEN flasche(levelnr)=  [3137]
flasche(levelnr)+1
3960 NEXT x,y                                [548]
3970 POKE (35000+(levelnr-1)*628),pokal(le  [3792]
velnr)
3980 POKE (35000+(levelnr-1)*628)-1,flasch  [2678]
e(levelnr)
3990 items(levelnr)=pokal(levelnr)+flasche  [3214]
(levelnr)
4000 IF items(levelnr)>255 THEN LOCATE 26,  [8643]
12:PRINT"TOO MUCH ITEMS.":LOCATE 26,14:PRIN
T"REDEFINE LEVEL.":a$="":WHILE a$="":a$=IN
KEY$:WEND:GOTO 3090
4010 POKE (35000+(levelnr-1)*628)-2,items(  [2850]
levelnr)
4020 POKE 40700+levelnr,1                    [895]
4030 LOCATE 28,12:PRINT"LEVEL STORED."      [2372]
4040 LOCATE 28,14:PRINT"GO ON ..."          [1110]
4050 a$="":WHILE a$="":a$=INKEY$:WEND       [1812]
4060 GOTO 3090                               [630]
4070 '                                       [117]
4080 '======================== save ====    [4024]
==================================
4090 '                                       [117]
4100 WINDOW #1,26,40,1,25:CLS#1:PEN 1       [2042]
4110 LOCATE 27,2:PRINT"PRESS"               [1118]
4120 LOCATE 27,4:PRINT"PLAY AND REC"        [1469]
4130 LOCATE 27,6:PRINT"THEN SPACE."         [2124]
4140 LOCATE 27,8:PRINT"-->"                 [1447]
4150 WHILE INKEY(47)<>0:WEND                [1642]
4160 LOCATE 32,8:PRINT"O.K."                [1465]
4170 LOCATE 27,11:PRINT"PLEASE WAIT."       [2036]
4180 LOCATE 27,23:PRINT"SAVING ..."         [1843]
4190 SAVE "!",b,34998,6002:LOCATE 27,18:PR  [4941]
INT"GO ON ...":CALL &BB03:CALL &BB06:GOTO
3010
4200 '                                       [117]
4210 '===================== load ======     [5805]
==================================
4220 '                                       [117]
4230 WINDOW#1,26,40,1,25:CLS#1:PEN 1        [2042]
4240 LOCATE 27,2:PRINT"PRESS PLAY"          [1636]
4250 LOCATE 27,4:PRINT"THEN SPACE."         [1847]
4260 LOCATE 27,6:PRINT"-->"                 [1288]
4270 WHILE INKEY(47)<>0:WEND                [1642]
4280 LOCATE 32,6:PRINT"O.K."                 [580]
4290 LOCATE 27,11:PRINT"PLEASE WAIT."       [2036]
4300 LOCATE 27,24:PRINT"LOADING ..."        [1929]
4310 LOAD"!",34998:LOCATE 27,18:PRINT"GO O  [4641]
N ...":CALL &BB03:CALL &BB06:GOTO 3010
4320 '                                       [117]
4330 '============================== play   [5708]
===================================
4340 '                                       [117]
4350 WINDOW#1,26,40,1,25:CLS#1:shape$(5)=C  [2741]
HR$(22)+"0"+" "
4360 LOCATE 27,2:PRINT"WHICH LEVEL"         [1308]
4370 LOCATE 27,4:PRINT"TO PLAY ?"           [1263]
4380 LOCATE 27,6:PRINT"-->"                 [1288]
4390 levelnr$="":WHILE levelnr$="":levelnr  [2423]
$=INKEY$:WEND
4400 levelnr=VAL(levelnr$):IF levelnr<1 OR [4230]
 levelnr>9 THEN CLS#1:GOTO 4360
4410 IF PEEK(40700+levelnr)<>1 THEN LOCATE  [9545]
 27,12:PRINT"I DON'T KNOW":LOCATE 27,14:PR
INT"THIS LEVEL.":LOCATE 27,20:PRINT"GO ON
...":CALL &BB06:CLS#1:GOTO 3200
4420 LOCATE 32,6:PRINT levelnr               [885]
4430 FOR y=1 TO 25                          [1112]
4440 FOR x=1 TO 25                           [768]
4450 feld(x,y)=(PEEK((35000+(levelnr-1)*62  [8189]
8)+((y-1)*25)+x)):LOCATE x,y:PRINT shape$(
PEEK((35000+(levelnr-1)*628)+((y-1)*25)+x)
)
4460 NEXT x,y                                [548]
4470 pokale=PEEK((35000+(levelnr-1)*628))   [1992]
4480 flaschen=PEEK((35000+(levelnr-1)*628)  [2680]
-1)
4490 items=PEEK((35000+(levelnr-1)*628)-2)  [3464]
4500 level=levelnr:items(level)=items       [2389]
4510 selfstyled=1:en=999:sc=0:it=0:CLS#1:G  [4117]
OTO 1990
4520 '                                       [117]
4530 ' ================= alle levels gesc    [3686]
hafft ==============================
4540 '                                       [117]
4550 MODE 1                                  [506]
4560 POKE &B28B,120                          [659]
4570 FOR a=1 TO 40:LOCATE a,1:PRINT shape$  [1559]
(0);:NEXT
4580 FOR a=1 TO 25:LOCATE 40,a:PRINT shape  [1855]
$(0);:NEXT
4590 FOR a=40 TO 1 STEP -1:LOCATE a,25:PRI  [3751]
NT shape$(0);:NEXT
4600 FOR a=25 TO 1 STEP -1:LOCATE 1,a:PRIN  [4240]
T shape$(0);:NEXT
4610 PEN 1:PRINT CHR$(22);"1"               [2584]
4620 FOR a=1 TO 40:LOCATE a,1:PRINT CHR$(1  [2260]
```

```
33);:NEXT
4630 FOR a=1 TO 25:LOCATE 40,a:PRINT CHR$( [2091]
133);:NEXT
4640 FOR a=40 TO 1 STEP -1:LOCATE a,25:PRI [3797]
NT CHR$(133);:NEXT
4650 FOR a=25 TO 1 STEP -1:LOCATE 1,a:PRIN [3513]
T CHR$(133);:NEXT
4660 PRINT CHR$(22);"0"                    [1295]
4670 LOCATE 5,3:PRINT"YOU HAVE COMPLETED A [3122]
LL 9 LEVELS."
4680 LOCATE 8,5:PRINT"THEREFORE THE GREAT  [1732]
ORTWIN"
4690 LOCATE 4,7:PRINT"WILL PROTECT ALL YOU [4434]
R LATER GAMES."
4700 IF sc<=hi THEN FOR a=1 TO 2000:NEXT:G [4609]
OTO 4760 ELSE hi=sc
4710 LOCATE 3,10:PRINT"ALSO YOU HAVE BROKE [3544]
N THE HIGH-SCORE."
4720 LOCATE 4,12:PRINT"SO YOU ARE WORTHY T [3776]
O BE KNOWN AS AN"
4730 LOCATE 14,14:PRINT"ORTWIN RUNNER."    [2449]
4740 LOCATE 9,16:PRINT"PLEASE ENTER YOUR N [2937]
AME :"
4750 LOCATE 9,20:CALL &BB03:INPUT">",ortwi [3735]
n$

4760 WINDOW#1,2,39,2,24:CLS#1              [1123]
4770 LOCATE 8,5:PRINT"TODAYS BEST ORTWIN R [1960]
UNNER:"
4780 LOCATE 12,10:PRINT UPPER$(ortwin$)    [2249]
4790 LOCATE 13,15:PRINT"WITH";hi;"POINTS"  [2786]
4800 '                                     [117]
4810 ' richard wagner  - brautchor aus 'lo [2298]
hengrin'
4820 '                                     [117]
4830 RESTORE 4890                          [757]
4840 ENV 3,10,-1,10                        [536]
4850 READ ton,dauer                        [1018]
4860 IF ton=-1 THEN FOR i=1 TO 3000:NEXT:G [3560]
OTO 990
4870 SOUND 1,ton/2,dauer,15,3:SOUND 2,ton* [5211]
2,dauer,15,3:SOUND 4,ton,dauer,15,3
4880 GOTO 4850                             [409]
4890 DATA 358,60,268,45,0,2,268,15,0,2,268 [7404]
,120,358,60,239,45,284,15,268,120,358,60,2
68,45,201,15,0,2,201,60,213,45,239,15,268,
60,239,2,268,2,284,45,268,15,239,120
4900 DATA 358,60,268,45,0,2,268,15,0,2,268 [6156]
,120,358,60,239,45,284,15,268,120,358,60,2
68,45,213,15,179,60,213,45,268,15,319,60,2
39,45,213,15,268,120
4910 DATA 358,60,268,45,0,2,268,15,0,2,268 [7404]
,120,358,60,239,45,284,15,268,120,358,60,2
68,45,201,15,0,2,201,60,213,45,239,15,268,
60,239,2,268,2,284,45,268,15,239,120
4920 DATA 358,60,268,45,0,2,268,15,0,2,268 [6371]
,120,358,60,239,45,284,15,268,120,358,60,2
68,45,213,15,179,60,213,45,268,15,319,60,2
39,45,213,15,268,120,-1,-1
4930 '                                     [117]
4940 ' franz schubert  -·die forelle       [2462]
4950 '                                     [117]
4960 ENV 2,10,-1,10                        [662]
4970 RESTORE 5010                          [924]
4980 READ ton,dauer:IF ton=-1 THEN RETURN  [6924]
ELSE IF ton=0 THEN SOUND 1,ton,dauer:SOUND
 2,ton,dauer:SOUND 4,ton,dauer:GOTO 4980
4990 dauer=dauer*1.5:SOUND 1,ton*2,dauer/2 [8290]
,15,2:SOUND 1,ton,dauer/2,15,2:SOUND 2,ton
,dauer,13,2:SOUND 4,ton*0.25,dauer,15,2
5000 GOTO 4980                             [423]
5010 DATA 426,25,319,25,0,2,319,25,253,25, [6335]
0,2,253,25,319,50,426,25,0,2,426,25,0,2,42
6,25,0,2,426,25,284,12,319,12,338,12,379,1
2,426,75,0,2
5020 DATA 426,25,319,25,0,2,319,25,253,25, [5555]
0,2,253,25,319,50,426,25,319,25,338,25,379
,12,338,12,319,25,451,25,426,75,0,2
5030 DATA 426,25,338,25,0,2,338,25,319,12, [5521]
338,12,379,12,338,12,319,50,426,25,319,25,
338,25,0,2,338,25,0,2,338,12,239,12,284,12
,338,12,319,75,0,2
5040 DATA 319,25,379,25,0,2,379,25,0,2,379 [5205]
,25,319,25,0,2,319,50,426,25,0,2,426,25,0,
2,426,25,0,2,426,25,284,25,338,25,319,75,0
,2
5050 DATA 319,25,379,25,0,2,379,25,0,2,379 [5385]
,25,319,25,0,2,319,50,426,25,0,2,426,25,0,
2,426,25,0,2,426,25,284,25,338,25,319,75,-
1,-1
```



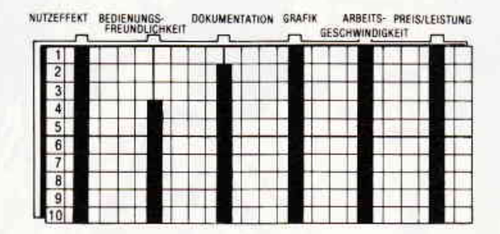
LASER BASIC

Hersteller: Ocean
Vertrieb: Computer Max
Steuerung: Tastatur
Monitor: Farbe/Grün
Programm: 100% M-Code
Preis: ca. 45,– DM

CPC 464 ☒ CPC 664 ☒ CPC 6128 ☒

Laser Basic ist im eigentlichen Sinne eine Basic-Erweiterung, die die grafischen Eigenschaften des CPCs erweitert. Allerdings geschieht dies in einem solchen Maße, daß man hier beim besten Willen nicht mehr von einer gewöhnlichen Erweiterung sprechen kann. Laser Basic wurde direkt zur Entwicklung von Spielprogrammen konzipiert und bietet hier mit über 200 neuen Befehlen einen so komplexen Leistungsumfang, daß man das Programm eher schon als Games-Designer bezeichnen kann.
Die 200 Befehle splitten sich auf in einen großen Block zur Spritedarstellung und bieten hier von Bewegungs- und Animationsroutinen über Vergrößerungsbefehle bis hin zu Kollisionsabfragen mit allem, was sich am Bildschirm darstellen läßt, so ziemlich jede Möglichkeit, die man sich überhaupt vorstellen kann.
Da die Abarbeitung der Sprites direkt über Manipulationen der Bitmap erfolgt, tritt auch bei 20 gleichzeitig am Bildschirm bewegten Sprites noch keine nennenswerte Zeitverzögerung oder Flimmern auf. Ein komfortabler Sprite-Editor erleichtert das Erstellen dieser Grafiken in beliebiger Größe und Farbwahl.
Die restlichen Befehle ermöglichen die bekannten Bildschirmmanipulationen wie Banking, Splits, Mehrfachmodes usw. Außerdem gibt es noch Befehle zur Interruptsteuerung von Programmen und Tonfolgen. Hat man einmal alle Interrupts des Basic gestartet, so kann man am Bildschirm im Direktmodus 20 animierte Sprites in mehreren Modes bewegen und dazu noch eine tolle Hintergrundmusik laufen lassen, ohne ein einziges Mal RUN eingegeben zu haben.
Laser Basic scheint tatsächlich das ultimative Tool für den Spieleprogrammierer zu sein. Neben dem eigentlichen Basic ist gleich noch ein passender Compiler für die neuen und alten Befehle des CPC erhältlich, mit dem man aus seinen Laser Basic-Programmen, Stand-Alone-Spiele machen kann, die die eigentliche Erweiterung nicht mehr benötigen, frei von Copyrights sind und vermarktet werden können. Bei Ocean in Arbeit befindet sich noch ein Screendesigner und ein Soundeditor, mit denen der Umgang mit dem ganzen System noch vereinfacht werden soll.
Allerdings hat eine so große Anzahl von Befehlen auch einen riesigen Nachteil. Um nicht unnötigen Speicherplatz zu verschwenden, hat man sich bei den Befehlsnamen auf drei Buchstaben-Kommandos beschränkt. Bei 200 Befehlen, die alle MSP, LSP, CLI oder SOO heißen, kann selbst ein versierter Programmierer ins Schwitzen kommen, da sich von den Kommandoformen auf die Auswirkung des Befehls keinerlei Schluß mehr ziehen läßt. Aber es gibt ja ein 150-seitiges deutsches(!!) Handbuch, welches über die Grundlagen der Spieleprogrammierung und die Anwendung von Sprites im allgemeinen informiert.
Bei einer derartig guten und kompletten Befehlssammlung dürfte in den nächsten Monaten mit einem Boom auf dem Spielesektor zu rechnen sein und so manche Spielidee, die im Hinterkopf eines Programmierers herumschwirrte und bislang in Ermangelung der nötigen Programmierkenntnisse ein Schattendasein fristete, bekommt nun doch noch ihre Chance, irgendwann einmal über den Bildschirm zu flimmern.

(TM)



VECTOR

Hersteller: GFA
Vertrieb: GFA
Steuerung: Tastatur
Monitor: Farbe/Grün
Programm: Basic/M-Code
Preis auf Anfrage

CPC 464 ☒ CPC 664 ☒ CPC 6128 ☒

Dreidimensionale Grafiken, die in der Gitterstruktur am Bildschirm dargestellt werden, haben eine Faszination, der sich kein echter Computerfreak entziehen kann.
Fangen sich diese Grafiken dann auch noch an zu drehen und zu bewegen, wie beispielsweise die Raumschiffe in Spielen wie Elite oder Starion, so geraten die Fans völlig aus dem Häuschen.
Vektorgrafik ist eines der faszinierendsten Themen, auf dem Gebiet der Computeranimation. Leider lassen sich diese rotierenden Gebilde sehr schlecht in Eigenarbeit programmieren. In Basic läuft da gar nichts.
Bei einer echten Vektorgrafik wird nach jeder gezogenen Linie die nächste errechnet, gezeichnet und die alte gelöscht. Mit nur einer Linie läßt sich hierbei auch schon ein annehmbares Ergebnis erzielen. Kommen jedoch mehrere Dimensionen hinzu und die Anzahl der zu zeichnenden Linien übersteigt die Zahl 3, so verlangsamt sich das Berechnen der neuen Linie den Zeichenvorgang so sehr, daß eine ruck- und flimmerfreie Darstellung nicht mehr möglich ist. Abhilfe schafft hier nur die Programmierung in Maschinensprache oder, wenn man es so einfach wie möglich haben will, das Programm VECTOR der Firma GFA.

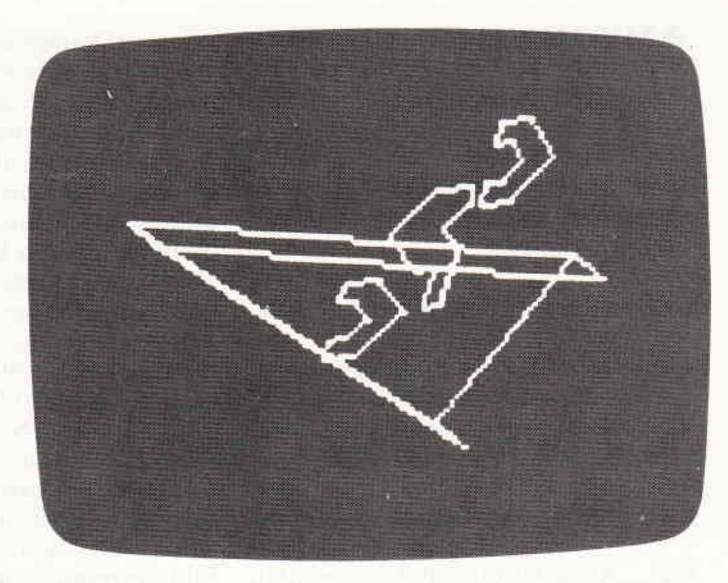
Geschrieben hat das Programm kein Geringerer als der CPC-Grafik-Spezialist J. Abel, der sein Können schon bei der Entwicklung des Profi Painters unter Beweis stellte.
VECTOR hat nichts mehr mit herkömmlichen Grafikprogrammen gemein. Nicht die Darstellung möglichst präziser 2D-Zeichnungen, sondern echte Bewegung am Bildschirm ist hier gefragt.
Um ein dreidimensionales Objekt am Bildschirm zu bewegen, müssen erst einmal die Form und die Eckdaten der Grafik festgelegt werden.
Dies kann bei VECTOR über zwei Wege geschehen. Zum einen kann das Objekt auf Papier in einem Koordinatensystem entworfen werden und die Daten werden dann einfach in tabellarischer Form eingegeben, oder man entwirft direkt am Bildschirm mit einem 3D-Grafikeditor. Zur Handhabung des Grafikeditors ist ein klein wenig räumliches Denken und ein Verstehen der Grundlagen der 3D-Grafik vonnöten. Ein gesondertes Kapitel im Handbuch vermittelt dieses Grundwissen jedoch auf eine sehr klare und verständliche Weise.
Ist ein Objekt fertig definiert, wird es in die Objektbibliothek aufgenommen und kann von hieraus abgespei-

chert, ausgedruckt, editiert und gezeigt werden. Die Option "Objekt Zeigen" bringt dem Anwender auch gleich das erste Erfolgserlebnis. Mit fünf Tasten kann das Objekt kontinuierlich um alle drei Achsen rotiert werden und stufenlos vergrößert und verkleinert werden. Dabei sind 255 Eckpunkte mit den entsprechenden Verbindungen möglich. Selbstverständlich ist man beim Anlegen von Objekten nicht nur auf eins beschränkt.

Nachdem man sich nun an dem drehenden und rotierenden Ding sattgesehen hat, fragt man sich, wo nun eigentlich der Einsatzzweck dieser Grafiken ist. Nun tritt die, meiner Meinung nach, stärkste Option des Vectorprogramms in Arbeit. Alle erzeugten Objekte können in einer Binärdatei, die gleichzeitig auch noch die nötigen RSX-Befehle zur Animation der Objekte enthält, abgelegt werden und somit in eigenen Basic-Programmen untergebracht werden.

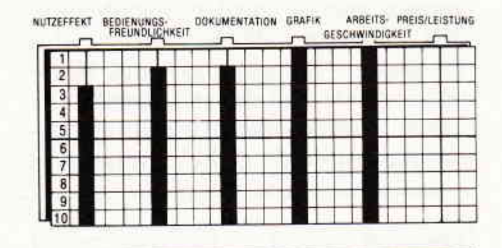
Damit sind dem Anwender nur noch Grenzen durch seine eigene Phantasie gesetzt. Ob nun zu Demo-Zwecken, als Ladebild, als Bestandteil eigener Spiele oder bei 2D-Darstellung als Sprite-Ersatz – hier stehen alle Möglichkeiten offen. Als zusätzlichen Bonus bekommt der Spieleprogrammierer noch die Möglichkeit, 16 interruptgesteuerte "Schüsse", die völlig unabhängig vom Basic-Programm laufen, in seine Programme zu implementieren.

Hier sehen Sie einmal das CPC-Logo von hinten. Mit dem Vector Grafik Editor recht schnell und einfach erstellt und danach im Zeige-Modus um alle Achsen gedreht. Leider läßt sich hier die wunderschöne, flackerfreie Bewegung nicht darstellen. Aber wir wollten Ihnen auf diese Weise einmal beantworten, wie das mittlerweile bekannte Emblem aus einer neuen Perspektive aussieht.



VECTOR ist eine völlig neue Programmidee und stellt eine Versuchung dar, der sich kaum ein Computeranwender widersetzen kann. Wer sich einmal in aller Ruhe die ca.10-minütige Demo des Programms angesehen hat, wird bass erstaunt sein, zu welchen grafischen Gewaltakten sein "kleiner" 8-Biter fähig ist.

(TM)



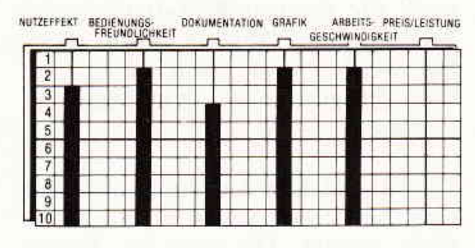
AMX-Utilities

Hersteller: AMX
Vertrieb: AMX
Steuerung: Maus
Monitor: Farbe/Grün
Programm: Basic/M-Code
Preis auf Anfrage

CPC 464 ☒ CPC 664 ☒ CPC 6128 ☒

Auf sehr schöne Art und Weise zeigt die Firma Advanced Memory Systems, daß sie die Käufer ihres Maussystems nicht auf sich gestellt lassen, sondern weiterhin mit qualitativ guter Software versorgen.

Das AMX-Utilities-Paket enthält einen Screeneditor, einen komfortablen Druckertreiber und ein Slide-Show-Programm.
Mit dem Screeneditor können Grafiken, die mit dem im AMX-Mauspaket enthaltenen ART-Programm erstellt wurden, nachbehandelt werden. Dazu bietet das Programm Spiegelungs- und Rotationsbefehle, mit denen entweder die ganze Grafik oder Ausschnitte gescrollt, gespiegelt, gedreht, vergrößert oder verkleinert werden.
Außerdem steht eine RAM-Disk zur Verfügung, auf der Bildschirmausschnitte gespeichert werden können und sich danach als Pinsel definieren lassen. Selbstverständlich erfolgt die komplette Steuerung über Ikon-Symbole mit der Maus.
Das Grafikprogramm orientiert sich dabei an Grafiksystemen, wie sie auf den 16-Bit-Rechnern der letzten Generation üblich sind, und kann (zwar bei niedrigerer Auflösung) den Bedienungskomfort recht gut emulieren.
Die Zoom- und Shrink-Option, die eine Vergrößerung oder Verkleinerung von Bildschirmausschnitten erlaubt, ist besonders interessant. Bildschirmausschnitte können wahlweise um die X- oder Y-Achse vergrößert oder verkleinert werden.
Der neue Druckertreiber ermöglicht Hardcopys auf nahezu allen Druckern im DIN-A4-Format. Auch hier erfolgt die Steuerung über Ikon- und Pull-Down-Menues. Die Hardcopy geht dabei recht schnell vonstatten.
Eine nette Option ist die Slide-Show, mit der erstellte Grafiken kontinuierlich am Bildschirm ausgegeben werden können. Die Reihenfolge der Grafiken ist frei wählbar.
Um mit der Utilities-Diskette arbeiten zu können, muß zunächst die Option MAKEDISC aufgerufen werden, die alle benötigten Programme der Master-Diskette und der Utilities-Diskette auf eine Workdisc überträgt. So ist der Anwender sicher vor Programmverlust und hat alle Anwendungen auf einem Datenträger versammelt.
Das Utilities-Paket ist eine schöne Idee, auf die sicherlich schon so mancher Anwender gewartet hat. Daß AMX hiermit noch nicht am Ende ist, beweist der Pagemaker sowie ein Soundeditor, die sich noch in Arbeit befinden, von deren Demos die englischen Fachmagazine schon in höchsten Tönen schwärmen.
Sobald diese Programme in Deutschland verfügbar sind, werden wir darüber berichten. (TM)

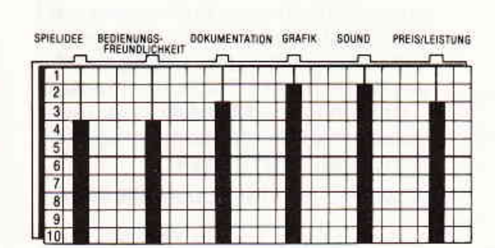




HEAVY ON THE MAGICK

Hersteller: Gargoyle Games
Vertrieb: Fachhandel
Steuerung: Tastatur
Monitor: Farbe/Grün
Autor: Greg Follis
Programm: 100% M-Code
Preis: ca. 30,– DM

CPC 464 ☒ CPC 664 ☒ CPC 6128 ☒



Durch eine leicht mißglückte Beschwörung wurde Axil der Fähige in ein seltsames Labyrinth versetzt. Unter den Ruinen der alten Burg Collodons Pile findet sich Axil in einem Raum wieder, in dem sich ein uraltes Zauberbuch befindet. Für Axil, einen hoffnungsvollen Nachwuchs-Magicker, eine hochinteressante Entdeckung.
Natürlich wird das gefundene Magie-Nachschlagewerk erst einmal gründlichst untersucht. Was Axil aber zu diesem Zeitpunkt noch nicht weiß, bringt ihn in tödliche Gefahr. Das Buch war nämlich durch einen besonderen Zauber geschützt.
Nachdem es von seinem Platz entfernt wurde, sind alle Monster, Geister und Ungeheuer, denen das unterirdische Labyrinth als Behausung dient, hinter dem ruchlosen Dieb her, um ihm so schnell wie möglich den Garaus zu machen. Alles in allem hat Axil eine ganze Menge Probleme.
Der einzige, der ihm bei der Bewältigung helfen kann, ist der Spieler, der sich dieses gelungene Spiel in seinen Rechner einlädt.
Mit "Heavy on the Magick" präsentiert das englische Softwarehaus Gargoyle Games ein in vielerlei Hinsicht neuartiges Programm.
Die Abenteuer von Axil dem Fähigen sind in gewisser Weise eine Mischung aus Abenteuer und Rollenspiel. Die auftretenden Monster und Axil verfügen über gewisse Charaktereigenschaften, aus diesen lassen sich gewisse Fähigkeiten, z.B. die Zauberei, ablesen. Einerseits gilt es, durch Herumlaufen und Untersuchen des Labyrinths, den Ausgang zu finden, andererseits gibt es dort aber eine ganze Reihe von Türen, die sich nicht so ohne weiteres öffnen lassen. Um dort hindurch zu gelangen, ist es notwendig, die für die jeweilige Tür geltenden Paßwörter zu kennen oder in Erfahrung zu bringen. Zu diesem Zweck ist es ratsam, sich an eines der herumstreunenden Monster zu wenden und sich ein wenig mit ihm zu unterhalten. Einer der besten Informationslieferanten ist Apex der Oger. Dieser wird Ihnen sicherlich schon sehr bald über den Weg laufen.
In Punkto Grafik hat "Heavy on the Magick" ebenfalls einiges vorzuweisen. Obwohl die Auflösung recht grob ist, gelang es den Programmierern, Beachtliches auf die Beine zu stellen.
Ob nun Axil oder Apex der Oger über den Bildschirm marschiert, ob sich eine Riesenvieper zum Angriff zusammenschlängelt, man hat jedesmal beim Auftreten einer animierten Figur den Eindruck, einem Zeichentrickfilm zuzusehen. Auch vom Sound her ist einiges Positives über dieses Spiel zu vermelden; zum Titelbild spielt das Programm ein wirklich gelungenes Musikstück, das der At-

mosphäre des Spieles gerecht wird. Im eigentlichen Spiel selbst sind dann zwar nur Geräuscheffekte enthalten, aber die erfüllen ihren Zweck voll und ganz.
Gargoyle Games hat in der dem Spiel beiliegenden Anleitung auf weitere Programme, in denen Axil der Fähige die Hauptrolle spielt, hingewiesen.
Nach den Qualitäten, die "Heavy on the Magick" vorzuweisen hat, darf man sicherlich auch auf die angekündigten Fortsetzungen gespannt sein.
(HS)



Unversehens findet sich Axil der Fähige in einem ausgedehnten unterirdischen Labyrinth wieder, das von einer Vielzahl seltsamer und unheimlicher Wesen bevölkert wird. Im, auf seine Weise, klassischen Gargoyle-Spielsystem gestaltet, ist HEAVY ON THE MAGICK eine der interessantesten Neuerscheinungen der letzten Tage. Action und Abenteuer geben sich hier ein Stelldichein.

WINTER GAMES

Hersteller: Epyx
Vertrieb: Fachhandel
Steuerung: Joystick
Monitor: Farbe/Grün
Programm: 100% M-Code
Preis: ca. 48,– DM

CPC 464 ☒ CPC 664 ☒ CPC 6128 ☒

Nun ist es endlich da! Das legendäre Sportspiel, das auf den Commodore-Computern neue Maßstäbe für Graphik und Spielbarkeit bei Computerspielen setzte. Durch den massiven Erfolg der CPC's wurde natürlich schon sehr bald an eine Umsetzung in Z-80-Code gedacht. Diese Tatsache allein ist recht löblich. Allerdings geht mit ihr eine Unsitte Hand in Hand, über die ich hier am besten einmal ein paar Worte verliere. Wer ab und zu die Anzeigen von Softwarehändlern durchgeht, dem dürfte sicherlich aufgefallen sein, daß WINTER GAMES seit gut einem halben Jahr beworben wird, allerdings erst seit einem Monat im Handel ist.
Große Softwarefirmen neigen dazu, Produkte schon lange vor der eigentlichen Fertigstellung anzupreisen. Bei den Schneiderrechnern ist diese unerfreuliche Tatsache zumeist im Zusammenhang mit Programmen, die erst von anderen Rechnertypen übernommen werden müssen, zu beobachten.
Mit wenig Überlegung kommt man sehr schnell zu dem Schluß, daß hier auf gewisse Art und Weise Vorablorbeeren einkassiert werden sollen.
Denn zumeist sind es Programme, die auf den Rechnern, auf denen sie ursprünglich implementiert wurden, enthusiastisch gefeiert wurden. Sei dies nun wegen der phantastischen



WINTER GAMES die Sportsimulation einer winterlichen Olympiade ist endlich auch für die CPC's erhältlich. Negativ fällt hier vor allem die bei der Kassettenversion nervtötende Ladezeit auf. Abgesehen davon, ist WINTERGAMES eigentlich das, was sich viele Sportfans erhofft hatten – eine gelungene und gut spielbare Sportsimulation.

Graphik oder der Animation oder Gott weiß warum. Sollen diese erstklassigen Effekte, die nur durch perfekte Programmierung erreicht werden können, nun von einem Prozessortyp auf einen anderen übertragen werden, und dabei auch nichts von ihrer Qualität verlieren, so versteht sich das von selbst, daß auch in diesem Falle ein nicht geringer Aufwand betrieben werden muß. Daß sich die Softwarehäuser in solchen Fällen mit der Ankündigung von Erstverkaufsterminen sehr gerne zu ihren Gunsten vertun (oder wie es scheint, sogar mit Absicht falsche Daten nennen) ist zwar gang und gäbe, sollte aber meines Erachtens in beiderseitigem Interesse abgestellt werden.

Nun ist WINTER GAMES aber ein Spiel, auf das sich das Warten wirklich gelohnt hat. Zwar ist das erste, was auffällt, die absolut ätzende Ladezeit der Kassette. Verschärft wird dieses Manko von der Tatsache, daß jeweils nur zwei Disziplinen gleichzeitig im Speicher des Rechners verbleiben. Möchte man nun also einmal komplett alle Disziplinen durchexerzieren, muß man sich, allein für eine Runde, auf eine Wartezeit von gut 20 Minuten einrichten. Aber es ist noch nicht aller Tage Abend, und eine Diskversion kommt bestimmt.

Die Disziplinen, die einem hier auf diesem winterlichen Computerszenario geboten werden, sind: Eiskunstlauf und Eiskunstlaufkür, Eisschnellauf, Skiartistik (Hot doging), Skispringen, Biathlon und Bobfahren. Jede dieser Disziplinen zeichnet sich durch ausgezeichnete Graphiken, sowohl Hinter- wie Vordergrund, eine ausgeklügelte Steuerung und passende Musik aus. Einige der Disziplinen, wie z.B. der Eiskunstlauf und Biathlon, dauern pro Durchgang ca. 5 Minuten. Andere wiederum nur wenige Sekunden (Skispringen). Kommt es nun vor, daß man die gesamten Winterspiele durchspielt, gestalten sich die Spiel- und Ladezeitintervalle sehr unterschiedlich, bisweilen nervtötend. Aber wie bereits erwähnt, sind das eigentliche nebensächliche Mängel, die bei einer Diskversion sicherlich nicht mehr ins Gewicht fallen. Alles in allem ist Wintergames ein Spiel, das sich am besten mit einigen Leuten zusammenspielen läßt. Rekordjagden machen auf diese Art und Weise ungemein viel Spaß. Es ist, so kann man mit Fug und Recht behaupten, wohl die beste Sportspielsimulation, die es bisher für die Schneiderrechner gibt.

(HS)

SPIELIDEE	BEDIENUNGS-FREUNDLICHKEIT	DOKUMENTATION	GRAFIK	SOUND	PREIS/LEISTUNG
3	3	3	1	3	2

TOMAHAWK

Hersteller: Digital Integration
Vertrieb: Fachhandel
Steuerung: Joystick + Tastatur
Programm: 100% M-Code
Preis: ca. 35,– DM

CPC 464 ☒ CPC 664 ☒ CPC 6128 ☒

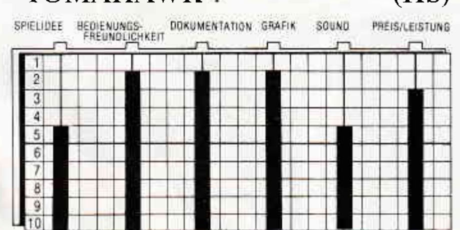
Nach wie vor erfreuen sich Flugsimulatoren aller Fasson einer wachsenden Beliebtheit. Eines der größten Hindernisse, die einem ergiebigen Spielerlebnis mit einer solchen Simulation im Wege stehen, ist die zumeist verwirrende Fülle von Bedienungskomponenten. Einerseits resultiert diese Vielzahl von Instrumenten aus den Bestrebungen der Programmierer, die Simulation so realistisch wie möglich zu gestalten, andererseits der Unfähigkeit, diese Bedienungselemente in eine ergometrische, bedienbare Form zu bringen. Ehrlich gesagt, hat mir dieser Sachverhalt das Spielen solcher Flugsimulationen meistens vergellt. Denn Tatsache ist, daß ein durchschnittlich lernfähiger Mensch meist zwei bis vier Wochen braucht, bis ihm ein fehlerfreier Flug gelingt. Sicherlich können Sie nun verstehen, daß ich auch "TOMAHAWK" anfangs recht skeptisch gegenüber stand. Zwar wird hier mit einem Hubschrauber geflogen, aber letztendlich ändert das nichts daran, daß es sich auch hier um einen Flugsimulator handelt. An meiner Skepsis änderte sich selbst nach der Lektüre der recht umfangreichen Bedienungsanleitung des Spiel nicht viel, denn bereits in der kurzen Tastenbelegungsreferenz wird auf siebzehn verschiedene Steuertasten hingewiesen. Zwar lassen sich vier dieser Steuertasten durch den Joystick ersetzen, aber in der Praxis zeigte sich dann, daß es besser ist, von vornherein nur auf der Tastatur zu spielen – Stick und Tasten sind zu verwirrend. Nachdem das Programm in den Rechner eingeladen war und ich die Lenslook-Prozedur, mit der das Spiel kopiergeschützt ist, vollzogen hatte, meldete sich "TOMAHAWK" zuerst einmal mit einem Menue, in dem sich gewisse Grundkomponenten seines Flugszenarios einstellen kann. Vier verschiedene Schwierigkeitsgrade lassen sich durch das Einstellen von Seitenwind-Turbulenzen und bewölkter Wetterlage variieren. Logischerweise erfolgten meine ersten Testflüge bei klarem Himmel, minimalem Seitenwind und auf der Spielstufe "Lernender". Zuerst einmal muß der Pilot, um seinen Hubschrauber zu starten, den Vergaser öffnen und die Neigung der Rotorblätter verändern, dadurch erhält der Hubschrauber den zum Start notwendigen Auftrieb. Ist es dem Spieler dann derart gelungen, seinen Hubschrauber in die Luft zu bekommen, und bei Gas und Rotorneigung einen halbwegs akzeptablen Wert anzulegen, ist es ab hier nur noch selten notwendig, an diesen beiden Instrumenten etwas zu verändern. Mit den Cursortasten läßt sich der Hubschrauber beschleunigen, verlangsamen und lenken. Da es in der Natur des "TOMAHAWK" liegt, ein Kampfhubschrauber zu sein, sind auch in dieses Spiel Luft-Luft und Luft-Boden Kampfhandlungen integriert worden. Dem Spieler stehen zu diesem Zweck drei verschiedene


ÜBER 380 SCREENS
Mit blitzschnellem Bildschirmaufbau.
In futuristischem High-Tech-Design.
SPINDIZZY ist eine künstliche Welt, ein riesiges Puzzle aus steilen Abhängen, engen Korridoren, hohen Wänden, schmalen Kanten und plötzlich auftauchenden Schluchten, durch die man seinen Weg finden muß. Mit System und ohne herunterzupurzeln.
Sinn und Zweck der Übung:
Die Abteilung für Geheimprojekte hat eine künstliche Welt entdeckt, die irgendwo im Raum schwebt. Und Ihre Aufgabe ist es nun, eine exakte Landkarte anzufertigen. Allerdings verfügen Sie nur über ein sehr altes Raumschiff, das ungeheuer viel Energie verbraucht, so daß Sie darauf angewiesen sind, sich auf dem künstlichen Planeten selbst zu versorgen.
Ein harter Job unter Zeitdruck, bei dem man häufig auf schier unüberwindliche Hindernisse trifft.
Und: Die Lösung liegt im 386. Screen!
Ein besonderer Leckerbissen: Die
MULTI-PERSPECTIVE
Funktion, mit der man die Hindernisse aus verschiedenen Blickwinkeln betrachten kann.
Spindizzy
A 3-D GAME OF SKILL
BY PAUL SHIRLEY
AMSTRAD®/SCHNEIDER® CASSETTE
Electric Dreams
SOFTWARE
Electric Dreams
Erhältlich als Cassette + Diskette für Schneider CPC und Commodore 64, als Cassette für Sinclair Spectrum.
Activision Deutschland GmbH, Postfach 76 06 80, 2000 Hamburg 76.
VERTRIEB DEUTSCHLAND:
Ariolasoft (Exclusiv-Distributor)
Rushware (Autorisierter Mitvertrieb)
VERTRIEB ÖSTERREICH:
Karasoft (Exclusiv-Distributor)
VERTRIEB SCHWEIZ
HILCU (Exclusiv-Distributor)



TOMAHAWK
Mit bißchen Übung kann der geneigte Spieler mit diesem rundherum gelungenen Flugsimulator schon nach kurzer Eingewöhnungszeit tolle Ausflüge unternehmen. Daß die Fliegerrei auch nach dem Beherrschen der Instrumente nicht langweilig wird, dafür sorgen verschiedene Optionen, durch die unter anderen die Wetterverhältnisse verschärft werden können.

Waffensysteme zur Verfügung, allerdings ist in diesem Zusammenhang anzumerken, daß man diese nicht ohne weiteres zum Einsatz bringen kann. Zuerst einmal müssen die Rotorblätter in eine eigens dafür vorgesehene Position gebracht werden. Erstaunlicherweise gelang es mir bereits beim zweiten Versuch, einen recht passablen Flug auf die Beine zu stellen. Ein Grund dafür ist sicherlich das erstklassige Hintergrundszenario, durch das man dann mit dem Hubschrauber fliegt. Mit wirklich toll gemachten Vektorgrafiken dargestellt präsentieren sich Berge, Flug-Zeughallen, Wälder, Hochspannungsleitungen und noch einer ganzen Menge mehr, in der Landschaft die man überfliegt.
Mit "TOMAHAWK" liegt ein Flugsimulator vor, dessen hervorstechendste Merkmale die verblüffend einfache Handhabung, die ausgezeichnete Spielbarkeit und die gelungene Grafik sind. Aber auch hier gilt: was sich leicht erlernen läßt, bedarf trotz allem einiger Übung bis zur Meisterschaft. Denn, nur die Tapfersten riskieren einen Flug mit der "TOMAHAWK". (HS)

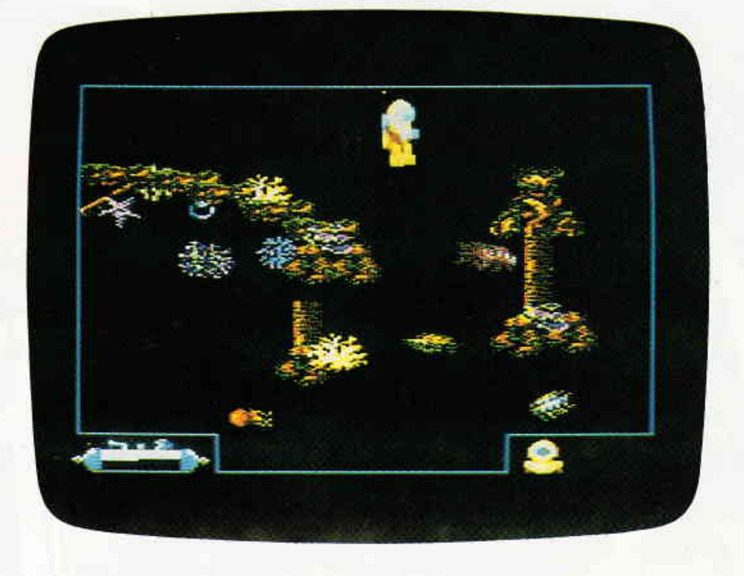


PACIFIC

Hersteller: ERE Informatique/
Gepo Soft
Vertrieb: Fachhandel
Steuerung: Joystick/Tastatur
Monitor: Farbe/Grün
Programm: 100 % M-Code
Preis: ca. 40,- DM

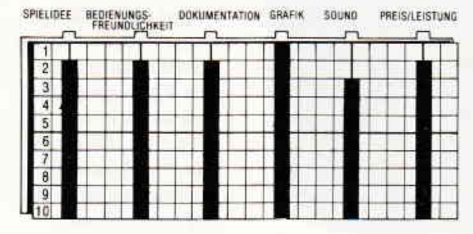
CPC 464 ☒ CPC 664 ☒ CPC 6128 ☒

Laut Homer, dem alten Griechendichter, lag die sagenumwobene Stadt Atlantis westlich der Säulen der Herkules. Heute sind diese Säulen eher unter dem Namen "Straße von Gibraltar" bekannt. Also müßten die Ruinen der versunkenen Metropole irgendwo in den Tiefen des Atlantiks zu finden sein. Aber wie das nun einmal mit Sagen und Legenden so ist, legt sich keiner den Zwang an, sie für bare Münze zu nehmen. So verlegte der Autor von "Pacific" die legendenumrankte Stadt in den Stillen Ozean, eben in den Pazifik. Ziel des Spieles, das mit seiner guten Multicolor-Grafik locker der Nachfolger von Sorcery werden könnte, ist es, die Ruinen von Atlantis zu finden. Ausgestattet wie ein Helmtaucher aus den Zwanziger Jahren mutet unser kleiner Unterwasserheld eher wie ein Astronaut an. Allerdings wimmelt es in den Pazifischen Gewässern regelrecht von gefährlichen Fischen. Eine einzige Berührung reicht aus, um eines seiner drei Leben zu verlieren. Ihren Anfang nimmt die abenteuerliche Tauchexpedition dicht unter der Wasseroberfläche, in einer kleinen Ausstiegsluke. Von dort aus geht es durch die oberen Zonen des Meeres. Immer tiefer sinkt unser Taucher in die geheimnisumwitterten Regionen des Ozeans. Schon bald beginnt das einfallende Sonnenlicht zu verblassen, im dunklen Wasser erscheinen die ersten Korallenriffe, die anfangs noch recht aufgelockert vorkommen, aber mit zunehmender Tiefe immer dichter und undurchlässiger werden. Schon bald findet man versprengte Zeugnisse der uralten Kultur, die hier ihr Ende fand. Im trüben Wasser tauchen Säulen und Obelisken auf, die zweifelsohne von Menschen erbaut wurden. Offensichtlich muß es schon früher einige Expeditionen in diesen Gewässern gegeben haben, denn immer wieder findet man im Gewirr der Riffe versteckte Luftpumpen.
Dies ist für unseren Taucher nur von Vorteil, kann er doch an diesen Lufttankstellen seinen Luftvorrat ergänzen. Aber nicht nur friedliche Wissenschaftler haben das Unterwasserareal erkundet, immer wieder trifft man auf Riffzonen, die vermient sind. Berührt man eine dieser Sprengkapseln, ist es aus, und ein weiteres Leben ist dahin. Außer dem Geheimnis Atlantis, das hier unter den Wellen des Pazifiks verborgen liegt, scheint es ein weiteres Mysterium zu geben, denn – ebenfalls im Riff versteckt – unser mutiger Taucher stolpert über eine ganze Reihe von Türen, die so aussehen, als haben sie einmal zu einem Schiff gehört. Durch diese Türen gelangt man dann in das Innere eines Wracks, dessen äußere Abmessungen von den alles überwuchernden Korallen verborgen werden. Findet man die richtige Türe, so kann man dort einige interessante Entdeckungen machen, aber es gibt viele Eingänge. Bei den meisten hat



In dieser phantasievoll und farbenfroh gestalteten Unterwasserwelt ist das neueste Spiel von Ere Informatique. Als Tiefseetaucher hat der Spieler hier die Aufgabe, die Geheimnisse des Pazifischen Ozeans zu ergründen. Die gelungene Grafik zeichnet PACIFIC als einen potentiellen Nachfolger des Nr.1 Spiels "Sorcery" aus.

man keinen Erfolg, sie entpuppen sich als Sackgassen.
Inzwischen müßte sich die Kunde der erstklassigen französischen Software herumgesprochen haben. Auch "Pacific" ist ein Produkt der Firma ERE Informatique, die uns schon mit CRAFTON & XUNC und EDEN BLUES in Erstaunen versetzt haben. So ist auch das vorliegende Programm ein echtes Kleinod. Die gelungene Farbgrafik läßt beim Vordringen in immer tiefer gelegene Regionen ein echtes "20.000 Meilen unter dem Meer"-Feeling aufkommen. Das Spielfeld, in dem die ganze Geschichte stattfindet, ist von enormen Ausmaßen. Obwohl ich "Pacific" eigentlich recht häufig spiele, glaube ich, daß dort noch eine ganze Reihe von Dingen sind, die ich bisher noch nicht erkundet habe. Bei der Orientierung in dem gewaltigen Unterwasserlabyrinth helfen die ab und zu herumliegenden Sextanten. Berührt man einen solchen und drückt dann den Feuerknopf, blendet das Programm eine Karte ein, auf der der eigene Standort verzeichnet ist. Auch gibt es Barrieren, die ein weiteres Vordringen unmöglich machen. Um hier weiterzukommen, muß man sich zuerst einmal den Weg freisprengen. Dies kann mit Pulverfässern geschehen, die gelegentlich herumliegen.
"Pacific" ist ein rundherum empfehlenswertes Spiel, an dem man lange seine Freude haben kann. Das enorme Spielareal, die tolle Grafik und die Geheimnisse, die es zu lösen gilt, werden jeden Spieler immer wieder auf "Pacific" zurückkommen lassen. (HS)

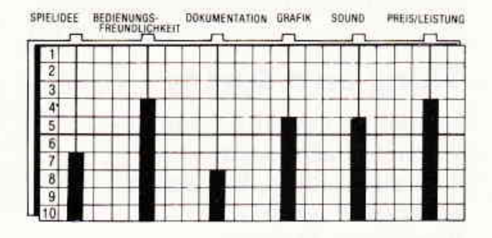


The Battle beyond the Stars

Hersteller: Solar Software
Vertrieb: Fachhandel
Steuerung: Tastatur/Joystick
Monitor: Farbe
Programm: 100% M-Code
Preis: ca. 34,- DM

CPC 464 ☒ CPC 664 ☒ CPC 6128 ☒

Ein wenig steinzeitlich wirkt es schon, was sich da auf dem Bildschirm abspielt. Wie Kamikazeflieger stürzen sich ganze Kaskaden von grauslichen, eroberungssüchtigen Aliens auf die letzte schützende Bastion, die die Erde verteidigt. Und derjenige, dessen Aufgabe hier die Steuerung der Verteidigungsanlage ist, sind natürlich Sie, der Spieler. Zweifelsohne ist "Battle beyond the Stars" altbekannten Arcadenhits wie "Galaxions" oder "Galaga" nachempfunden. Als Homecomputerversion für den Schneider liegt hiermit als so etwas wie ein Novum vor. Wenn man sich gerne mit derartigen "Zap em up"-Spielen beschäftigt, dann findet der geneigte Spieler hier mehr als genug Action.
Eine "Dicker Daumen"-Garantie (durch permanentes Feuerknopf betätigen) kann im vorliegenden Fall mit ruhigem Gewissen ausgesprochen werden. Bei der Erstellung des Programms haben die Autoren sich sicherlich viel Mühe gegeben. Es ist sozusagen ein gelungenes Spiel geworden. Daß die Spielidee dabei





"Battle beyond the Stars" fällt auf den ersten Blick durch die etwas zu kurz gekommene Spielidee auf. Denn grundsätzlich geht es nur darum, Aliens, die sich in immer neuen Wellen kamikazeartig auf das Raumschiff des Spielers stürzen, zu zerstören. Wer sich damit zufrieden gibt, kommt zweifelsohne auf seine Kosten.

doch ein wenig zu kurz gekommen ist, liegt wohl eher an der Vorlage, die man sich gesucht hat, denn am Programm kann nichts bemängelt werden. Nichtsdemzutrotze ist "Battle beyond the Stars" ein nettes und gut spielbares Programm geworden.
(HS)

Thing on a Spring

Hersteller: Gremlin Graphics
Vertrieb: Fachhandel
Steuerung: Joystick/Tastatur
Monitor: Farbe/Grün
Programm: M-Code
Preis: ca. 30,– DM

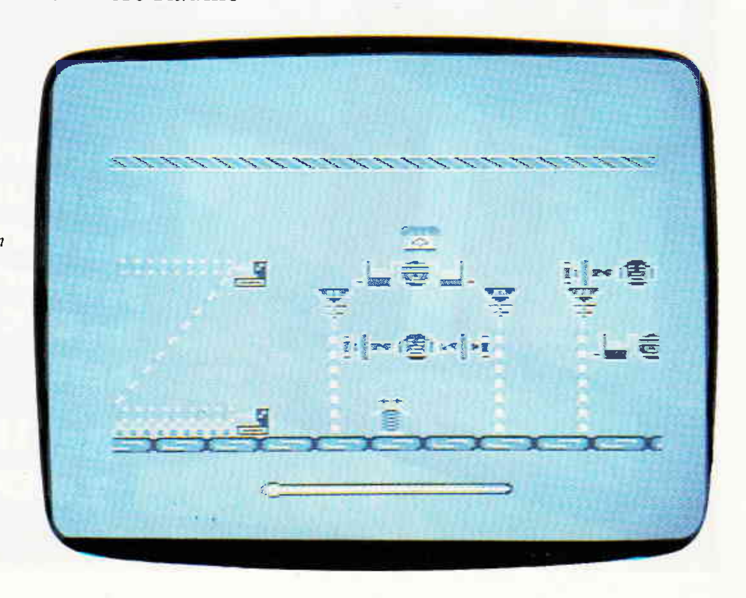
CPC 464 ☒ CPC 664 ☒ CPC 6128 ☒

Magie und Technologie stellen den Hintergrund dar, in dem sich der Held dieses lustigen Vertreters der Arcade-Spiele wiederfindet. Hat doch ein übler Kobold all seine Magie darauf verwendet, sich ein Mittel zur Eroberung der Welt zu erschaffen. Und man glaubt es kaum, aber scheinbar eignet sich für dieses Unterfangen eine moderne Technologie um einiges besser als althergebrachte Magie mit ihren doch schon leicht eingestaubten Zaubersprüchen. So sieht sich Kumpan Kobold auch gezwungen, der baldigen Verwirklichung seines Planes wegen, all sein Hab und Gut (und noch das seiner leicht murrenden Spießgesellen) in bare Münze zu verwandeln. Aber nicht mittels eines genialen Zauberspruchs, sondern durch die Nutzung einer viel mächtigeren und böseren Gewalt – der schrecklichen, gnadenlosen und verheerenden freien Marktwirtschaft.

Als er schließlich das benötigte Geld beisammen hat, besorgt er sich eine Fabrik; genauer gesagt eine Waffenschmiede. Diese versetzt er, nun aber unter Zuhilfenahme seiner magischen Fähigkeiten, tief in den Bauch der Erde, in eine unbekannte, gewaltige Höhle. Aber schon bald taucht der Held dieser Geschichte auf; eine seltsame Mischung aus grünem Männlein vom Mars und einer ausgeleierten Bettfeder (zumindest hört es sich so an). Kaum überblickt er die gesamte Situation, entschließt er sich aus reiner Nächstenliebe, ohne auch nur einen Gedanken an die etwaige, sicherlich nicht bescheidene Belohnung zu verschwenden, dem bösen Kobold Mores zu lehren und seine Fabrik stillzulegen.

In gewisser Weise ist man ein wenig an das alte Manic-Miner-Spielkonzept erinnert, wenn man sich "Thing on a Spring" zum ersten Mal anschaut. Sicherlich kommt das nicht von ungefähr, denn auch hier muß man auf Plattformen rumlaufen, umherhüpfen und Gegenstände einsammeln. Aber damit hat es sich dann auch schon mit den Gemeinsamkeiten. Die Fabrik des Kobolds teilt sich in vier Stockwerke auf, in denen man mittels eines Aufzuges "umher-sprungfedern" kann. Jede Etage hat vier große Räume, die durch Korridore miteinander verbunden sind. Diese Räume scrollen horizontal über den Monitor. Zwar hat unsere mutige Sprungfeder gegenüber Normalsterblichen den Riesenvorteil einer großen Sprungkraft, aber um sich diese zu erhalten, benötigt sie immer wieder Öl. Aber glücklicherweise sind außer den eigentlich einzusammelnden Puzzleteilen noch eine ganze Reihe Ölfässer in der ganzen Fabrik verteilt. Mit deren Hilfe ist es überhaupt erst möglich, die gestellte Aufgabe zu erfüllen und, nachdem alle Puzzleteile geholt wurden, dem bösen Kobold eins auszuwischen. Auf dem erwähnten Puzzle ist nämlich ein Tip verborgen. Ein Tip, der dem Spieler zeigt, was er tun muß, um die gestellte Aufgabe zu erfüllen. Alles in allem hat THING alles, was man von einem gutem Spiel erwartet. Ganz besonders angetan hat es mir der Sound. Mit derartig guten Musikuntermalungen wird man auf dem CPC ja wirklich nicht verwöhnt, und was da bei THING aus dem Lautsprecher tönt, ist wirklich nicht von schlechten Eltern. Aber nicht nur durch den Sound hebt sich THING aus der grauen Softwaremasse hervor. Es ist am Anfang nicht so schwierig, so daß auch diejenigen, die keine Joystickakrobaten sind, akzeptable Ergebnisse erzielen können. Nach einigen Räumen wird es dann allerdings schwieriger. Laserstrahlen, verschwindende Plattformen und Schalter, die einen bei Berührung an die unmöglichsten Plätze versetzen, tragen dazu bei, daß dieses Spiel auch nach einiger Zeit noch nicht langweilig wird.
(HS)

SPIELIDEE · BEDIENUNGS-FREUNDLICHKEIT · DOKUMENTATION · GRAFIK · SOUND · PREIS/LEISTUNG
1 2 3 4 5 6 7 8 9 10



Tief unter der Erde liegt die Fabrik, mit der der böse Kobold und seine Spießgesellen die Herrschaft über die Erde erringen wollen. Der einzige, der zur Stelle ist, um dies zu verhindern, ist das Sprungfederding. In vier verschiedenen Stockwerken gilt es, Puzzleteile und Geschenkpackungen einzusammeln. Ohne diese Dinge ist es nicht möglich, die Macht des Kobolds zu brechen

Fremdmonitore am CPC



Welche Monitore passen? Was bedeutet RGB oder composite video Anschluß? Große Unsicherheit wird beim Anwender sowie auch in der Beratungsszene beim Monitor oder Computerkauf offenbart. Paßt der Monitor auch an andere Systeme? Kann ein günstig angebotener Farbmonitor eine Alternative zum Schneider-Farbmonitor sein? Kauf eines Modulators oder eines Farbmonitors? Eine falsche Entscheidung führt hier oft zu Enttäuschungen und Geldverlust.

Das falsche Datensichtgerät kann auch einen negativen Einfluß auf das Wohlbefinden einer am Computer tätigen Person haben, was sich dann durch frühzeitige Ermüdungserscheinungen, Kopfschmerzen und anderes unangenehm bemerkbar macht.

Das Ziel dieses Artikels soll es sein, Hinweise zu geben, wann ein Monitoreinsatz sinnvoll ist, und über die Vermittlung einiger Grundlagen, die auftretenden Fragen im Zusammenhang mit dem Anschluß eines Fremdmonitors am CPC zu beantworten. Ein Schwerpunkt liegt also bei der Schnittstelle Computer - Monitor. Die Darstellung einiger Grundlagen aus der Bildübertragungstechnik wird viele Fragestellungen von selbst beantworten, und ist nützlich bei Problemlösungen im gesamten Bereich der Bilddarstellung über Monitore.

Die Gliederung dieses Artikels:
1. Was ist ein Monitor und wozu dient er?
2. Wann ist die Anschaffung eines Monitors sinnvoll (Modulator plus TV oder Monitor)?
3. Wie arbeitet ein Monitor (s/w und Farbe)?
4. Welche Schnittstellen gibt es?
5. Die Schnittstelle des CPC
6. Fazit und Tips

Was ist ein Monitor?

Ein Monitor ist ein Gerät, um dem Menschen Informationen, die in elektrischen Signalen enthalten sind, schnell erfaßbar darzustellen. Er wandelt also elektrische Signale in optische um. Prinzipiell ist es dem Monitor egal, von wem er die Signale zur Darstellung erhält. Man kann ihn als ein universell einsetzbares Peripheriegerät bezeichnen, siehe hierzu Bild 1.
Ob am:
- Computer als Ausgabegerät für Daten
- Videorecorder zum Betrachten von Filmen
- TV/Tuner zum Konsumieren eines Fernsehprogramms
- Videokamerasystem zur Überwachung von Räumen,

Mit einem einzigen Monitor könnte man vom Prinzip her alle diese Aufgaben meistern. Voraussetzung ist, daß der Monitor seine zur Darstellung benötigten elektrischen Informationen immer in der gleichen Art angeboten bekommt. Hier liegt ein Problem (siehe später: Schnittstellenproblem).

Wann ist die Anschaffung eines Monitors sinnvoll?

Dieses Thema soll im Hinblick auf Nutzung eines Monitors im Bereich der Schneider CPC Computertechnik erörtert werden. Eine Ausdehnung auf andere Systeme würde den Rahmen dieses Artikels sprengen. Das Prinzip der gemachten Aussagen läßt sich jedoch ohne weiteres in andere Bereiche übertragen.
Dank des lobenswerten Konzepts der Firma Schneider, Komplettsysteme kostengünstig, sofort einsetzbar und als Grundsystem schon mit einem Monitor (grün oder Farbe) ausgerüstet, zu vertreiben, befindet sich ein CPC-Besitzer in den meisten Anwendungssituationen in einer deutlich besseren Lage als dies der größte Teil von Computereigentümern anderer Hersteller ist. Es ergeben sich zwei Ausgangssituationen:
A) Man ist Eigner eines CPC mit Farbmonitor. Ein zusätzlicher monochromer Monitor hat eigentlich nur für denjenigen Nutzen, der sich häufig für viele Stunden mit Anwendungsprogrammen wie Dateiverwaltungen, Textverarbeitungen, Tabellenkalkulationen und anderen Arbeitsprogrammen (meistens im 80 Zeichen-Modus) beschäftigt. Obwohl der Schneider Farb-Monitor eine ausgezeichnete Darstellungsqualität liefert, die auch in obengenannten Anwendungsbereichen einen Teil der Benutzer zufriedenstellen wird, erreicht man mit einem monochromen Monitor eine noch ruhigere und klarere Darstellung von Zeichen. Ein direkter Sichtvergleich beim Computerhändler muß letztendlich das ausschlaggebende Argument liefern, ob sich für Sie diese Investition lohnt. Achten Sie beim Vergleich auf die Detaildarstellung von Buchstaben im 80 Zeichen-Modus. Weitere Auswahlkriterien siehe Punkt.6.
B) Man ist Eigentümer eines CPC mit grünem Monitor. Dies wird sicherlich auf den größeren Teil der CPC-Eigentümer zutreffen. Die Frage einer nachträglichen Anschaffung eines Farbmonitors wird hier am häufigsten gestellt. Die zusätzliche Darstellung von Farben ist in einigen Bereichen auch wirklich ein deutlicher Gewinn.
Farbmonitore sind relativ teuer. Die angebotenen Modulatoren, die einen Anschluß des Computers an ein normales Farbfernsehgerät ermöglichen, können für einige Bereiche eine deutlich preiswertere Alternative sein. Ein Farbfernseher steht heute in vielen Wohnzimmern. Diesen Weg zu einer Farbdarstellung würde ich allerdings nur denjenigen uneingeschränkt empfehlen, die diesen nur in größeren Zeitabständen nutzen wollen, um mit ein paar Freunden zu spielen. Der Spiele-Freak aber fährt sicherlich besser mit einem Farbmonitor. Bei denjenigen, die sich in allen nur denkbaren Bereichen mit dem Computer auseinandersetzen, überwiegt häufig das Arbeiten mit farbiger Grafik. Bei häufigem Grafikeinsatz sollte die Anschaffung eines Farbmonitors in Erwägung gezogen werden. Hat ein Programmierer es im we-

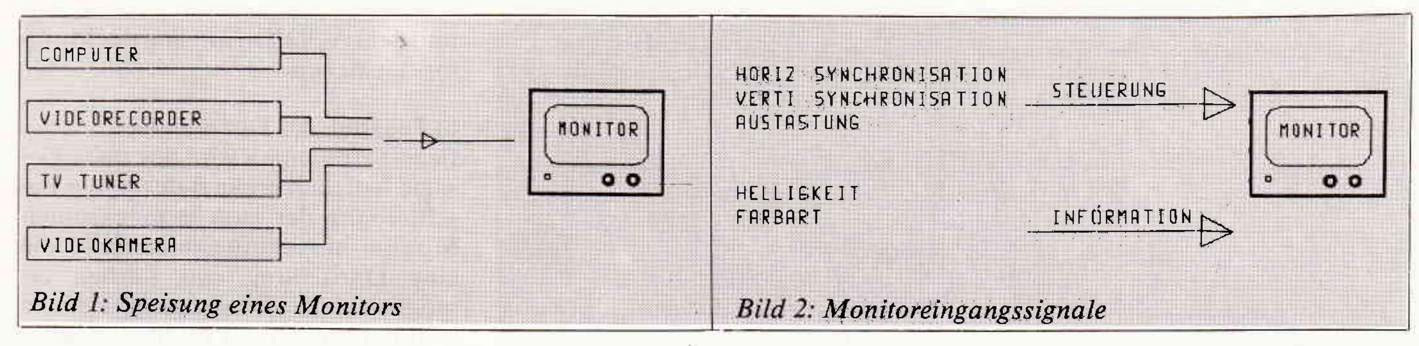


Bild 1: Speisung eines Monitors

Bild 2: Monitoreingangssignale

sentlichen mit Aufgabenstellungen zu tun, bei denen die farbige Darstellung eine untergeordnete Rolle spielt, so kann der Modulator eingesetzt werden, um (in der Schlußphase der Programmierung) die Farben einzugeben. Bei CPC-Anwendern, die Arbeitsprogramme verwenden, in denen die farbige Darstellung wichtig ist, gibt es zum Farbmonitor keine Alternative.
Außer einer klareren Darstellung ergibt sich beim Farbmonitor gegenüber dem Fernseher der Vorteil einer dem Abstand zum Benutzer gerechteren Bildschirmgröße. Eine größere Nachleuchtdauer der Monitorbildröhre mildert Flimmereffekte.

Wie arbeitet ein Monitor?

Wir kennen zwei Arten von Monitoren:
a) Monitore, die nur ein einfarbiges Bild darstellen (auch bezeichnet als monochrome, grüne oder schwarzweiß-Monitore) und
b) Monitore, die mehrere Farben darstellen können.
Man könnte Monitore auch nach anderen Kriterien gruppieren, aber die obere Aufteilung ist die meistverwandte, auch in der Computer-Branche.
Anmerkung: Dieser Artikel behandelt Monitore, wie sie in der Fernseh-, Video- und Computertechnik verwendet werden (auch als Video-Monitore bezeichnet). Die zum Monitor angelieferten elektrischen Signale sind nach dem hier gültigen TV-Standard konzipiert, und der Monitor arbeitet ebenfalls nach den Spezifikationen dieser Norm (z.B. zeilenweise Darstellung in 625 Zeilen, 25 Bilder pro Sekunde). Monitore, die nach anderen Normen arbeiten, können nicht am CPC betrieben werden! Es gibt Monitore, die nach anderen Prinzipien laufen (z.B. x,y,z-Monitore). Diese werden hier nicht behandelt – haben sie doch keinerlei Bedeutung in der Computertechnik.
Ein Monitor braucht außer zugeführter Energie (Stromanschluß) verschiedene Eingangssignale. Hier unterscheidet man zwei Gruppen. Einmal die Signale, die eigentlich die darzustellende Information beinhalten, d.h. Helligkeits- oder Farbwerte als Funktion der Zeit. Die zweite Gruppe sind die Steuer-(Synchronisations)-signale, die der Monitor zur Darstellung der gelieferten Informationen im richtigen Format benötigt; denn der Monitor muß wissen, welches die erste Information eines gesendeten Bildes oder einer Zeile ist, siehe hierzu Bild 2.
Damit sind wir schon mitten drin, in der Erklärung des Funktionsprinzips eines Monitors. Lassen wir zunächst die Farbe weg.

a) Funktion des grünen (monochromen) Monitors

Auf Bild 2 erkennen wir, als Eingangssignale werden die horizontalen und vertikalen Synchronisationssignale, sowie das die Information enthaltende Helligkeitssignal (Leuchtdichte, Luminanz) benötigt.
Alle diese Signale können zu einem Signal (-gemisch) vereint sein, und über einen Leiter (plus Bezugsleiter oder Erde), meist in Form eines Koaxialkabels, übertragen werden. Das Signal heißt BAS-Signal (composite video signal) und ist zusammengesetzt aus:
- dem Bildsignal B,
- dem Austastsignal A und
- den Synchronsignalen S.

Sehen wir uns Bild 3, das Blockschaltbild des monochromen Monitors einmal an und versuchen daraus herzuleiten, was mit unserem Eingangssignal passiert und wie seine Einzelkomponenten zum Bildaufbau im Monitor beitragen.

Der Monitor stellt die Informationen zeilenweise dar. Ein Elektronenstrahl wird durch magnetische Ablenkung zeilenweise über die Leuchtschicht der Bildröhre geführt. Je nach verwandter Leuchtschicht ergibt sich dadurch eine farbige (z.B. grüne) Leuchtspur. Die zwei Ablenkspulen (vertik./horiz.) erhalten ihre Ströme (ausschlaggebend für den Aufbau der ablenkenden Magnetfelder) durch die jeweiligen (vertik./horiz.) Generatorstufen, die synchronisiert durch die zugehörigen Synchronisationsimpulse, die notwendigen vertikalen und horizontalen Ablenkspannungen erzeugen. Diese beiden Synchronisationsimpulse werden von unserem eingespeisten BAS-Signal abgetrennt und in vertikale und horizontale Pulse aufgetrennt. Sie sorgen für die Synchronisation, d.h., daß der eine Zeile schreibende Elektronenstrahl an jedem Ort dieser Zeile die zugehörige Bildinformation (Helligkeit) enthält.
Die Helligkeitsinformation steckt in der Aufprallgeschwindigkeit der Elektronen auf die Leuchtschicht

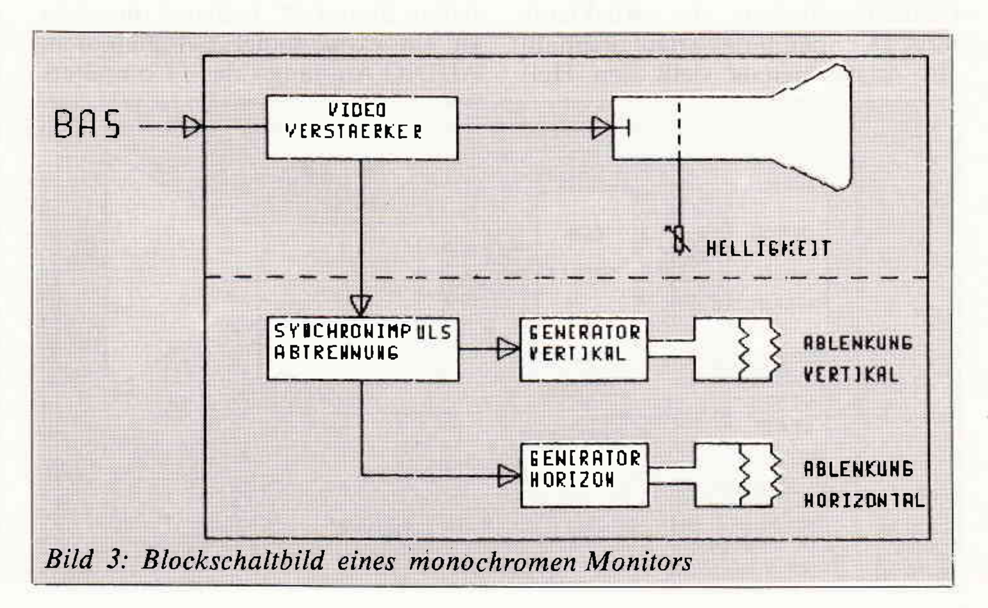


Bild 3: Blockschaltbild eines monochromen Monitors

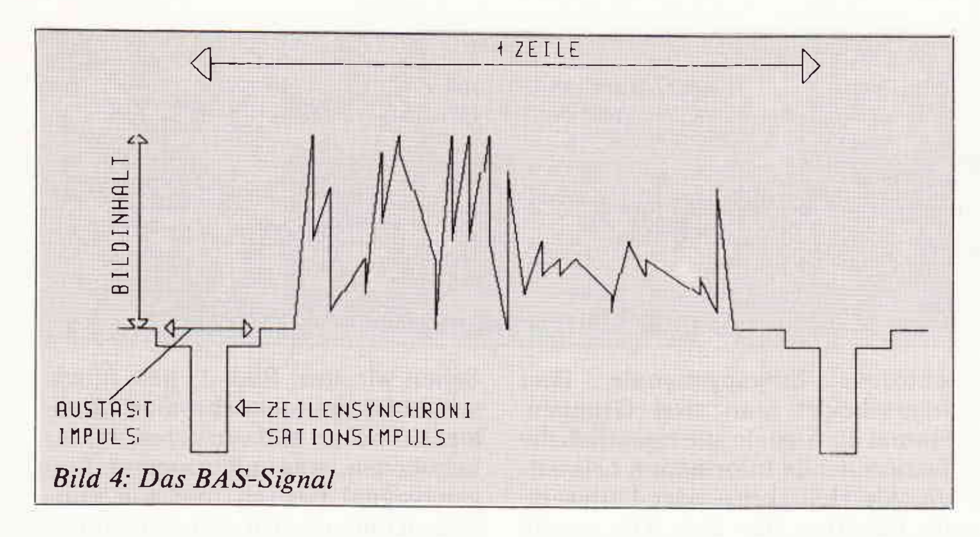


Bild 4: Das BAS-Signal

(kinetische Energie). Je größer die Energie, und je mehr Elektronen aufprallen, umso heller leuchtet der Aufprallpunkt.
Die Geschwindigkeit der Elektronen kann durch Spannungen an den Elektroden der Bildröhre beeinflußt werden. Über eine Gitterspannung wird meistens die Grundhelligkeit eingestellt, während die Bildinformation durch die Katodenspannung ihren Einfluß findet. Die Katode erhält das Bildsignal vom Videoverstärker.
Außer der Bildinformation enthält dieses Signal meist noch das Austastsignal, welches während der Strahlrückläufe (nach dem Schreiben einer Zeile eines Bildes) dafür sorgt, daß diese auf dem Bildschirm unsichtbar bleiben (Austasten).
Bild 4 zeigt den Aufbau des BAS-Signals

Wir können jetzt mit unseren Vorkenntnissen dieses Signal deuten. Der Zeilensynchronimpuls trennt die Zeilen (Abstand 64 us nach TV-Norm). Er initiiert den Strahlenrücklauf am Ende einer Zeile. Er liegt mitten im Austastimpuls, der für das Unsichtbarbleiben des Rücklaufs sorgt. Der Spannungsverlauf oberhalb des Austastsignals stellt den Helligkeitsverlauf einer Zeile dar. Jeder Punkt auf dieser Kurve stellt einen Helligkeitswert dar, der zu einer bestimmten Position auf dieser Zeile gehört.
Das einzige, was am BAS-Signal von Bild 4 fehlt, ist das vertikale Synchronisationssignal (Bildanfang). Da es nur alle 625/2 Zeilen auftritt (es werden zwei Halbbilder geschrieben), zudem länger als eine Zeile ist und aus mehreren Teilen besteht, habe ich es in Bild 4 weggelassen. Es ist zum weiteren Verständnis nicht wichtig, dieses Signal im Detail zu kennen. Der Monitor bekommt durch dieses Signal die Information: "die nächste Zeile, die kommt, ist die erste eines Bildes". Nun haben wir Eingangssignal und Funktionsprinzip des grünen Monitors beim Schneider CPC kennengelernt (Anschlußdetails siehe Schnittstelle).
Anmerkung: Einige Monitore sind noch mit einem Tonteil ausgestattet. Dieses wäre im Bild 3 ein separater Teil mit den Blöcken Verstärker und Lautsprecher. Zu dem Eingangssignal BAS würde es ein zweites Eingangssignal AUDIO geben, welches die niederfrequente Toninformation liefert. Beim CPC bedeutet dieses, es ist ein extra Kabel zwischen dem AUDIO-Ausgang des Computers und dem AUDIO-Eingang des Monitors zu ziehen.

b) Funktion des Farbmonitors
Beim Farbmonitor wird das zuvor Gesagte lediglich ergänzt durch die Beschreibung der zum monochromen Monitor hinzuzufügenden Teile, die eine Darstellung von Farben ermöglichen. Die Eingangssignale zum Monitor müssen als Bildinformation jetzt zusätzlich zur Helligkeit (Luminanz- oder Leuchtdichtesignal) die Farbinformation (Farbartsignal F) enthalten. Das Farbartsignal wird auch Chrominanz genannt. Wir erhalten also als Eingangssignal für das BAS-Signal beim monochromen Monitor das FBAS-Signal beim Farbmonitor.
Während wir beim monochromen Monitor als Bildinformation die Helligkeitsverteilung ansahen, redet man beim Farbmonitor von der Farbverteilung. Der Begriff "Farbe" steht dann also für folgende Komponenten:

Farbe	
Leuchtdichte	**Farbart**
(Helligkeit) (Luminanz)	(Chrominanz)
Farbton (Wellenlänge)	Farbsättigung (Intensität)

Unser Eingangssignal zum Monitor muß also außer den Synchronisationssignalen und dem Austastsignal folgende drei Bildinformationen liefern:
i) Leuchtdichte
Den Einfluß dieser Größe kann man sich am besten vor Augen führen, wenn man an die Projektion eines Farbdias mit einer 100W Birne oder mit einer 300W Birne denkt.
Die Leuchtdichte des projizierten Bildes mit 300W ist höher. Denselben Einfluß kann man beim Farbmonitor sichtbar machen, wenn man die Helligkeit des Bildes am entsprechenden Regler verändert.
ii) Farbton
Das farbige Licht ist nichts anderes als eine elektromagnetische Welle mit einer bestimmten Wellenlänge (je nach Farbe). Der Farbton wird durch die Wellenlänge des Lichtes bestimmt, und ist damit das wesentlichste Unterscheidungsmerkmal der Farbe.
iii) Farbsättigung
Die Farbsättigung gibt den Weißanteil einer Farbe an. Wird z.B. Rot durch den Zusatz von Weiß entsättigt, so geht Rot in Rosa über. Eine Sättigung von 0 % entspricht Weiß, während eine 100 %-ige Sättigung der Spektralfarbe entspricht.

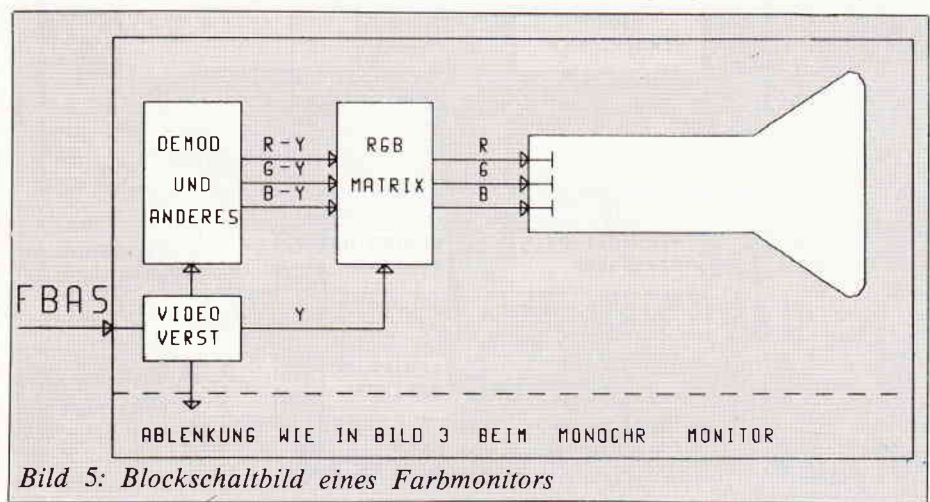


Bild 5: Blockschaltbild eines Farbmonitors

Vielen von uns ist aus der Physik bekannt, daß durch ein Prisma fallendes weißes Licht zerlegt wird in seine Spektralfarben. Ebenso kann aus der Summe der Spektralfarben wieder weißes Licht gewonnen werden. Hierzu reichen aber auch drei Farben wie Rot, Grün und Blau. Diese heißen dann Primärfarben. Durch additive Farbmischung dieser Primärfarben kann jeder beliebige Farbton erzeugt werden. Dieses Prinzip wird bei der Farbbildröhre angewendet.
Jeder Bildpunkt der Farbbildröhre muß jede Farbe darstellen können. Jeder Punkt besteht aus drei kleinen Leuchtstreifen in den Farben Rot, Grün und Blau. Es wird mit drei Elektronenstrahlen (drei Katoden) gearbeitet, wobei jeder Strahl nur auf die Leuchtstreifen seiner ihm zugeordneten Farbe trifft. Es gibt also eine Katode für Rot, eine für Grün und eine für Blau.
Anmerkung: Die hier vorgenommenen Funktionsbeschreibungen beziehen sich auf das Prinzip der RGB-Ansteuerung der Bildröhre an den Katoden. Die anderen Verfahren, wie RGB-Ansteuerung an den Gittern oder die Farbdifferenzansteuerung an Gittern bzw. Katoden seien hier der Vollständigkeit halber erwähnt.
Schauen wir uns jetzt das Blockschaltbild des Farbmonitors an.
Wir sehen in Bild 5, daß der gesamte Strahlsteuerungsteil genauso funktioniert wie beim monochromen Monitor. Lediglich im Signalteil haben wir Ergänzungen, deren Aufgabe es ist, sen beiden kann man (G-Z) zurückgewinnen.
Alle drei Farbdifferenzsignale und das Z-Signal werden der RGB-Matrix zugeführt, in der die Signale RGB zur Bildröhrenansteuerung gebildet werden.
Wir kennen jetzt das vereinfachte Funktionsprinzip des Farbmonitors aus dem FBAS-Signal die Bildinhaltsinformation so aufzubereiten, daß die drei Signale Rot, Grün und Blau entstehen, die dann direkt die Bildröhre steuern.

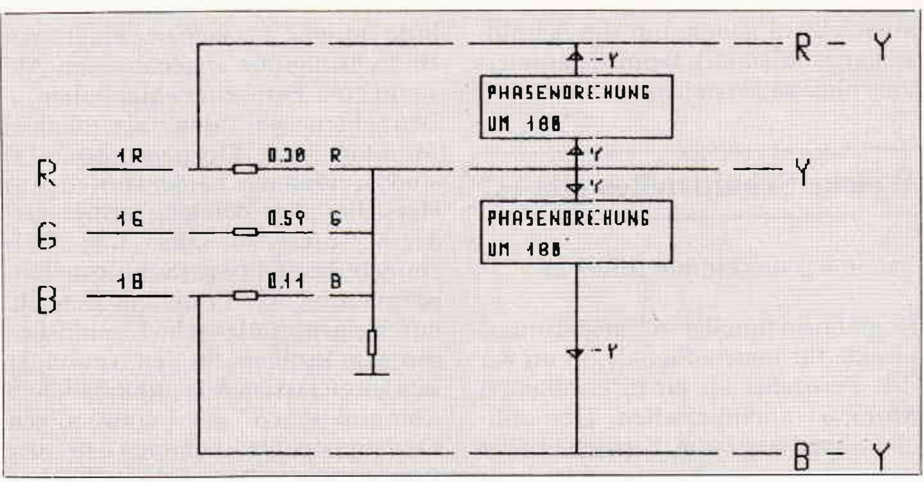


Bild 6: Erzeugung der Farbdifferenzsignale

Um zu verstehen, wie dieses im Monitor geschieht, müssen wir wissen, wie die Bildinformationen im FBAS-Signal enthalten sind. Einige Vereinfachungen seien erlaubt, um zum Verständnis der eigentlichen Thematik unnötige Details nicht allzu breit auszubreiten.
Die Herstellung des FBAS-Signals hält sich natürlich an unsere hier gültigen TV-Normen. Eine wichtige Eigenschaft muß dieses Signal mitbringen, Stichwort: Kompatibilität. Dieses Farbbildsignal muß, in einen monochromen Monitor eingespeist, eine normale einfarbige Darstellung ermöglichen. Hierzu werden die Ausgangssignale einer Farbfernsehkamera (Rot, Grün, Blau) dahingehend korrigiert, wie das Helligkeitsempfinden des menschlichen Auges auf diese drei Farben reagiert.
Das Leuchtdichtesignal, wie es im Farbmonitor benötigt wird, das aber ebenso beim Anschluß eines monochromen Monitors als Bildinformationssignal dient, wird daher nach folgender Formel hergestellt:

$$Y = 0,30\ R + 0,59\ G + 0,11\ B$$

Wie wir später sehen werden, wird im CPC das Lumineszenzsignal für den grünen Monitor entsprechend erzeugt. Um die Bandbreite des entstehenden FBAS-Signals klein zu halten, werden in der Fernsehtechnik neben dem Z-Signal nicht die drei Primärfarben Rot, Grün und Blau, sondern zwei Farbdifferenzsignale übertragen, und zwar (R-Z) und (B-Z). Zur Übertragung der Farbinformation werden also nur diese beiden Signale verwandt. Zu ihrer Erzeugung gibt Bild 6 nähere Informationen.

Diese beiden Signale werden dann je einem Farbträger aufmoduliert, addiert, und als Farbartsignal zusammen mit dem Z-Signal, den Synchron- und Austastsignalen zum FBAS-Signal summiert.
In unserem Farbmonitor werden aus diesem Eingangssignal wieder die Einzelsignale erzeugt. Die Synchronsignale werden abgetrennt und analog dem monochromen Monitor zur synchronen Ablenksteuerung der Elektronenstrahle benutzt. Das Z-Signal wird abgetrennt und der RGB-Matrix zugeführt. Das abgetrennte Farbartsignal wird demoduliert und in die beiden Farbdifferenzsignale (R-Z) und (B-Z) gewandelt. Aus die-

ausreichend genug, um die Schnittstellenproblematik beim Computeranschluß zu verstehen.

Welche Schnittstellen gibt es?

Sehen wir uns einmal Bild 7 an.

Es gibt prinzipiell drei Einspeisungspunkte für unsere Signale, wenn wir den Fernseher als einen erweiterten Monitor miteinbeziehen. Der übliche Signalweg beim Fernseher über den Antenneneingang bis zur Bildröhre ist der längste, d.h. er ist mit den meisten Bildverarbeitungsvorgängen behaftet, was zu Einbußen bei der Bildqualität führt. Die ursprünglichen Bildinformationssignale liegen, wie auch beim CPC, als RGB-Signale vor. Speisen wir diese über eine RGB-Schnittstelle in den Monitor, so haben wir den kürzesten Weg zwischen Erzeugung und Darstellung der Bildsignale. Hier ist es möglich, die beste Bildqualität zu erzeugen. Der Schneider CPC bietet uns diesen Weg, wie wir noch sehen werden, an.

Wollen wir einen Fernseher nutzen, so müssen die Signale im zuerworbenen Modulator über ein FBAS-Signal zu dem modulierten HF-Signal umgeformt werden, um in den Antennen-Eingang in dem vom Fernseher erwarteten Format eingespeist zu werden. Im Fernseher werden alle diese gemachten Umformungen wieder "rückwärts" vollzogen, damit am Ende wieder die ursprünglichen RGB-Signale zur Bildröhrenansteuerung vorliegen. Sie erkennen in Bild 7, welch ein langer Signalweg mit vielen Verarbeitungsvorgängen hier vorliegt, und wie leicht hier Verfälschungen der ursprünglichen Signale entstehen können.
Es gibt heute übrigens schon Fernseher, die einen RGB-Eingang besitzen. Bei Benutzung des Fernsehers bitte immer versuchen, einen der Bildschirmgröße angemessenen Abstand zum Fernseher einzuhalten.
Betrachten wir den eigentlichen Monitor: zwei Einspeisungspunkte sind zu erkennen. Leider verstehen es Hersteller aller Bereiche immer wieder hervorragend, selbst eine solch einfach herstellbare Schnittstellennormierung, wie in diesem Bereich, durch eigenbrödlerische Eigenlösungen zum Problem für den Anwender werden zu lassen. Alle nur möglichen Kombinationen der notwendigen Eingangssignale kommen in der Praxis vor. Die gebräuchlichen Schnittstellen sind in den Tabellen 2 und 3 abgebildet.

Bei der zweiten und der vierten Version kann der Synchronisationseingang noch aufgeteilt sein in je einen Eingang für vertikale und horizontale Synchronisation. Dieses würde dann noch einen zusätzlichen Leiter bedeuten.
Die erste Version ist die bedeutendste und sollte bei einem universell einsetzbaren Monitor unbedingt vorhanden sein. Beim Farbmonitor muß bei diesen vier Schnittstellen allerdings mit einem Abfall an Bildqualität gegenüber den nachfolgend beschriebenen RGB-Anschlüssen gerechnet werden (zu langer Signalweg: Siehe oben).
Alle drei Schnittstellen haben als gemeinsames Kennzeichen die drei Eingangssignale Rot, Grün und Blau. Eingespeist in einen Monitor werden sie fast direkt zur Steuerung der Bildröhre benutzt, was eine geringe Signalverfälschung, d.h. eine hohe Bildqualität gewährleistet.
Es gibt Unterschiede im Aussehen der Farbsignale zwischen der Analog- und der TTL-Schnittstelle. Die Bezeichnung TTL kommt aus dem Bereich der elektronischen Logikschaltungen (Transistor-Transistor-Logik). Hier wird nur mit den Signalen "wahr - unwahr" bzw. den diesen Zustände repräsentierenden Gegebenheiten, wie z.B. "Strom ein - Strom aus" oder "Signal 5 Volt - Signal 0 Volt", gearbeitet.
Bei unserer RGB-TTL-Schnittstelle bedeutet dies, die Farbsignale können nur zwei Zustände einnehmen, d.h. z.B. für das Signal Rot: "Rot ein - Rot aus" (5 V - 0 V). Jeder der drei Eingänge kann nur diese zwei Zustände einnehmen. Das heißt, es können sich acht verschiedene Eingangssituationen ergeben, oder anders ausgedrückt, es können acht verschiedene Farben dargestellt werden (die drei Primärfarben und ihre Mischfarben: rot, grün, blau, weiß, schwarz, gelb, cyan, purpur).
Beim RGB-Analog Eingang können die Farbsignale in einem bestimmten Bereich beliebige Werte annehmen, d.h. das Signal Rot kann stufenlos (analog) entsprechend verschiedener Leuchtdichten andere Werte darstellen. Die Zahl der verschiedenen Eingangssituationen ist unendlich. Es können alle Farben dargestellt werden.
Der RGBI-Eingang stellt eine Mischung aus den beiden anderen Versionen dar. Die Farbsignale RGB liegen als TTL-Signale vor. Um mehr Farben darzustellen, muß ein extra Intensitätssignal zum Monitor geführt werden. Bei allen drei Versionen ist es denkbar, die Synchronisationssignale über einen extra Eingang einzuspeisen, oder diese Signale mit über die RGB-Anschlüsse zu führen. Es kommt beides in der Praxis vor.

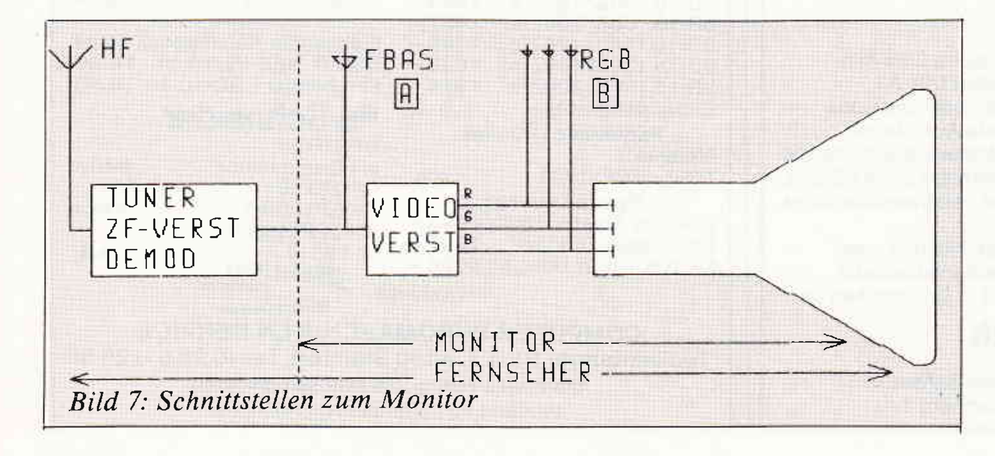
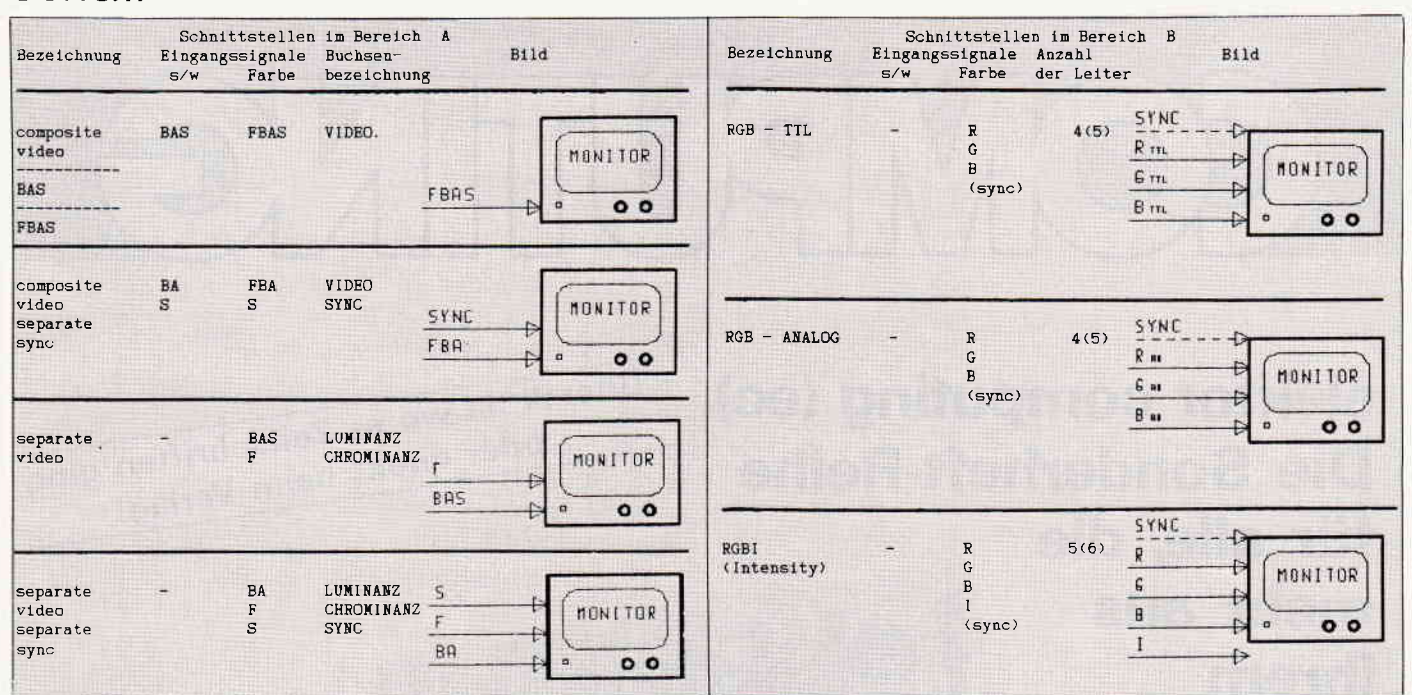


Bild 7: Schnittstellen zum Monitor

Die Schnittstelle des CPC

In Bild 8 sehen wir die Monitor-Schnittstelle des CPC.
Zum Anschluß eines Farbmonitors werden uns die drei Farbsignale R, G und B zur Verfügung gestellt, und zwar in analoger Form. Die noch benötigten Synchronisationssignale (vertik. u. horiz.) werden uns auf einer extra Leitung angeboten. Dieses ist also die RGB-ANALOG-Schnittstelle mit extra sync-Leitung.
Ein weiteres Ausgangssignal ist erkennbar. Dieses wird in den Handbüchern mit Leuchte oder Lumineszenz bezeichnet. Es dient zum Anschluß eines monochromen (grünen) Monitors. Aus den drei Farbsignalen wird ein etwa dem Helligkeitsempfinden des Auges für diese drei Farben entsprechendes Signal geformt, das durch Addition der entsprechenden Farbanteile entsteht. Die Synchronisationssignale werden hinzuge-



Bezeichnung	Schnittstellen im Bereich A – Eingangssignale s/w	Farbe	Buchsenbezeichnung	Bild
composite video BAS FBAS	BAS	FBAS	VIDEO.	FBAS → MONITOR
composite video separate sync	BA S	FBA S	VIDEO SYNC	SYNC, FBA → MONITOR
separate video	-	BAS F	LUMINANZ CHROMINANZ	F, BAS → MONITOR
separate video separate sync	-	BA F S	LUMINANZ CHROMINANZ SYNC	S, F, BA → MONITOR

Tabelle 2

Bezeichnung	Schnittstellen im Bereich B – Eingangssignale s/w	Farbe	Anzahl der Leiter	Bild
RGB - TTL	-	R G B (sync)	4(5)	SYNC, R TTL, G TTL, B TTL → MONITOR
RGB - ANALOG	-	R G B (sync)	4(5)	SYNC, R, G, B → MONITOR
RGBI (Intensity)	-	R G B I (sync)	5(6)	SYNC, R, G, B, I → MONITOR

Tabelle 3

mischt und es entsteht ein composite video Signal (BAS).
Dieses Signal ist ungeeignet zum Anschluß eines Farbmonitors. Hierzu benötigen wir, wie wir wissen, ein FBAS-Signal, dessen Entstehung vorab beschrieben wurde.

Fazit und Tips

Beim Anschluß eines Fremdmonitors darf man nicht vergessen, daß der Schneider-Monitor auch die Spannungsversorgung für den CPC enthält. Der Schneider-Monitor muß also immer noch mit dem CPC über die Spannungsversorgungseingänge verbunden und eingeschaltet sein, auch wenn man über den Monitoranschluß einen Fremdmonitor mit Bildsignalen versorgt. Besitzt man eine externe Spannungsversorgung, so entfällt diese Notwendigkeit.

Was ist also zu beachten?
Fall A)
CPC-Besitzer mit Farbmonitor: Der monochrome Fremdmonitor sollte eine composite video (BAS)-Schnittstelle besitzen. Die Schnittstelle composite video separate sync kann ebenfalls vom CPC versorgt werden.
Fall B)
CPC-Besitzer mit grünem Monitor: Der Farbmonitor als Fremdmonitor sollte über RGB-ANALOG Eingänge verfügen. Die Synchronisationssignale sollten über einen extra sync-Eingang eingespeist werden können.

Schaut man sich die Monitore in den Verkaufsregalen an, so sieht man die Eingangsbuchsen oft nur mit RGB bezeichnet. Fragen Sie hier nach, oder lassen Sie sich die Spezifikationen im Benutzerhandbuch zeigen. RGB-ANALOG wird auch als RGB-LINEAR, RGB-TTL als RGB-DIGITAL bezeichnet.
Schließen Sie Ihren CPC an einen Farbmonitor mit TTL-Eingängen an, so werden Sie zwar ein farbiges Bild bekommen, aber bei Überprüfung werden Sie feststellen, daß von den 16 Farben, die der CPC darstellen kann, nur 8 sichtbar sind.
Legen Sie Wert darauf, einen möglichst universell einsetzbaren Farbmonitor zu erwerben, dann sollte er folgende Eingänge besitzen:
a) composite video (FBAS)
b) separate video (Luminanz Chrominanz)
c) RGB-ANALOG
d) RGB-TTL
e) RGBI
Möchten Sie vor dem Erwerb eines Monitors außer dem Sichttest vielleicht die Spezifikationen mit anderen Monitoren vergleichen, so sind folgende Parameter wichtig:
a) Bandbreite (möglichst groß; zB.: 22 MHz)
b) Auflösung horiz. (möglichst groß; z.B.: 920 Punkte)
c) Auflösung vert. (möglichst groß; z.B.: 350 Punkte)
Als Maß für die Auflösung wird oft auch die Angabe gemacht, wieviel Linien aufgelöst werden oder einfach auch nur 80 Zeichendarstellung angegeben.
Einige Farbmonitore können auf Schalterdruck zur monochromen Bildwiedergabe umgeschaltet werden, was die Auflösung verbessern kann und den Einsatzbereich dieses Monitors erweitert.
Die Schnittstelle des Schneider CPC stellt uns also gängige Signale zur Verfügung, mit denen wir handelsübliche qualitativ hochwertige Monitore anschließen können. Beim Farbmonitor auf RGB-ANALOG Eingänge achten. Tauchen Probleme auf, so ist das Monitorverbindungskabel auf richtige Belegung der Steckeranschlüsse zu überprüfen. Die CPC-Anschlüsse siehe Handbuch; die Monitoranschlüsse sind aus der Monitordokumentation zu entnehmen. (Dipl.-Ing. V. Klug)

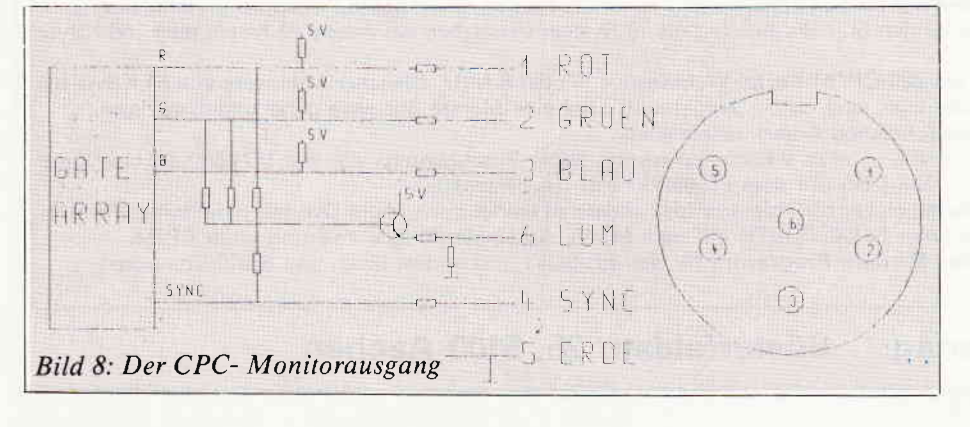
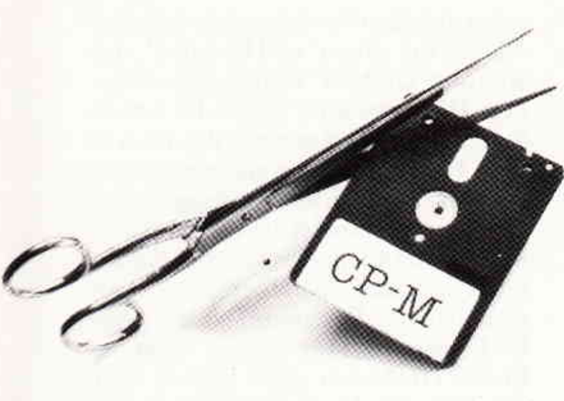


Bild 8: Der CPC- Monitorausgang

CP/M und seine Möglichkeiten



Wie in der vorangegangenen Folge bereits angekündigt, soll unser Thema auch dieses Mal wieder die Hochsprache FORTRAN sein. Nachdem beim letzten Mal der NEVADA FORTRAN-Compiler vorgestellt wurde, wollen wir in dieser Folge ein FORTRAN-Programm schreiben und einen Vergleich sowohl mit CBASIC als auch in der nächsten Folge mit PASCAL anstellen. Ausgangspunkt ist zunächst einmal das bereits in der Ausgabe Nr. 6 abgedruckte Listing einer FUNCTION in CBASIC.

Nichts geht ohne Editor

Beim Schreiben eines Programmes sind wir natürlich erst einmal wieder auf einen vernünftigen Editor angewiesen. Leider konnte sich SchneiderData bisher noch nicht dazu durchringen, NEVADA FORTRAN mit diesem wichtigen Werkzeug auszurüsten, so daß man auf ein eigenes Textverarbeitungsprogramm angewiesen ist, z.B. CONTEXT (s.a. CPC Nr. 4) oder den auch hier möglichen Umweg über den Schneider-BASIC eigenen Editor gehen muß. Ist man glücklicher Besitzer des HISOFT C-Compilers aus gleichem Hause, so kann man auch sehr gut den hier mitgelieferten WORDSTAR-kompatiblen ED80 bzw. ED82 einsetzen. Dieser Fullscreen-Editor hat nur eine Länge von ca. 8kB und enthält die wichtigsten Befehle zum Erstellen von Programmen. Anders als bei WORDSTAR wird der gesamte zu bearbeitende Text auf einmal in den Arbeitsspeicher geladen, wodurch zeitraubende Diskettenzugriffe entfallen. Dadurch kann ein Programmlisting z.B. wesentlich schneller gescrollt werden. Als sehr angenehm wird auch die Tatsache empfunden, daß der Cursor nicht mehr nachläuft und im Gegensatz zu WORDSTAR wieselflink nach rechts oder links bewegt werden kann.

Programmieren in FORTRAN

Jetzt soll es aber endlich losgehen: Unser kleines CBASIC-Programm, das wir nun in die Sprache FORTRAN umschreiben wollen, bestand aus einem sogenannten Hauptprogramm, dem wir den Namen MAIN gegeben hatten und einer FUNCTION mit Namen MULTI, die von MAIN aufgerufen wurde. Die FUNCTION hatte lediglich die Aufgabe, die drei von MAIN übergebenen Werte gemäß den drei enthaltenen arithmetischen Anweisungen zu multiplizieren und die Ergebnisse zurückzugeben. In einem typischen FORTRAN IV Listing würde das so aussehen:

```
1234567
-----------------------------------
      PROGRAM MAIN
C
C     EINLESEN DER 3 VARIABLEN E,F,G
C
      READ(1,1000) E,F,G
 1000 FORMAT(3(/,F10.5))
C
      WERT = MULTI (E,F,G)
C
C     AUSGABE DER BERECHNETEN WERTE
C
      WRITE(1,2000) E,F,G,WERT
 2000 FORMAT(4(/,F10.5))
C
      STOP
      END
C
      FUNCTION MULTI(A,B,C)
C
      A = A * A
      B = B * B
      C = B * A
      MULTI = C
C
      RETURN
      END
```

Wenn man sich dieses Listing etwas genauer ansieht, erkennt man wahrscheinlich eine gewisse Ähnlichkeit mit einem BASIC-Programm, die entwicklungsgeschichtlich bedingt ist.

Fangen wir einfach einmal vorne an.

Die Zahlen von 1 – 7 gehören nicht zum eigentlichen Listing, sondern dienen der Verdeutlichung einer ganz speziellen Eigenart von FORTRAN, wobei es ganz besonders auf die Spalten von 1 – 7 ankommt. Diese Spalten sind nur für bestimmte Anweisungsmarken (Labels) zugelassen, z.B. im Zusammenhang mit

einer FORMAT-Anweisung, die in dem obigen Programm gleich zweimal auftaucht. Für diese ersten 7 Spalten gelten sehr strenge Regelungen.
In der ersten Spalte darf z.B. nur ein C stehen. Ist dieses C gesetzt, so wird die ganze nachfolgende Zeile als Kommentar interpretiert, vergleichbar einem REM unter BASIC. Die eigentliche FORTRAN-Anweisung darf erst in Zeile 7 beginnen, wie z.B. die READ-Anweisung, die mit einem INPUT in BASIC vergleichbar ist. Da beim Entstehen der Sprache die Programm- und Dateneingabe nur über genau spezifizierte Lochkarten möglich war, erwartet FORTRAN IV auch heute noch bei der Eingabe von Daten eine genaue Angabe der Vor- und Nachkommastellen, was z.B. beim Einlesen von Diskettendaten sogar sehr nützlich sein kann. In unserem Fall erfolgt diese Spezifizierung durch die auf die READ-Anweisung folgende FORMAT-Anweisung, die über das Label 1000 zugeordnet wird. Dieses Label finden wir daher auch innerhalb der Klammer, nach dem Komma, in der vorangegangenen READ-Anweisung wieder. Über die 1 vor dem Komma wird außerdem noch ein bestimmter Eingabekanal zugeordnet, z.B. die Konsole. Der Inhalt der FORMAT-Anweisung "3(f10.5)" besagt, daß drei (F)loatingpoint-Variablen eingelesen werden sollen, die jeweils aus 10 Zeichen, einschließlich Dezimalpunkt, bestehen. Die Anzahl der Nachkommastellen beträgt genau 5. Wenn z.B. die Zahl 3 über die Konsole eingegeben werden soll, müssen jetzt aber nicht etwa die entsprechenden Vor- und Nachkommastellen mit Nullen aufgefüllt werden, der Dezimalpunkt hinter der 3, also "3.", darf jedoch auf keinen Fall vergessen werden. Weiterhin ist es mit dieser FORMAT-Anweisung nicht möglich, mehr als 5 Stellen nach bzw. mehr als 4 Stellen vor dem Komma korrekt einzulesen. Bei der Ausgabe, die durch das WRITE-Statement vergleichbar mit einem PRINT unter BASIC erfolgt und ebenfalls eine FORMAT-Anweisung benötigt, werden die entsprechenden Nachkommastellen mit Nullen ausgefüllt. Z.B. 4.00000. Für Anwender, denen diese Prozedur etwas zu umständlich erscheint, hält NEVADA FORTRAN auch wesentlich vereinfachte, an FORTRAN 77 angelehnte Ein-/Ausgabe-Statements bereit, die dann allerdings nicht dem ANSI 66 Standard entsprechen. Nun wollen wir aber endlich zum Aufruf der FUNCTION MULTI kommen.

FUNCTION unter FORTRAN

Unser Listing beginnt, wie in FORTRAN vorgeschrieben, mit dem Hauptprogramm MAIN. Dieses wird mit einem STOP und einem END beendet. Daran anschließend folgt die FUNCTION MULTI, die immer als eine selbständige Programmeinheit betrachtet wird und infolgedessen ebenfalls nach dem RETURN mit einem END abgeschlossen werden muß. Wenn Sie das Programm jetzt starten, geben Sie doch einfach wieder die Zahlen 2,3 und 4 für die Variablen e,f und g ein, damit wir unser Resultat mit der CBASIC-FUNCTION vergleichen können. Wenn Sie alles richtig gemacht haben, muß die Ausgabe lauten:
4.0000 9.00000 36.00000 36.00000

Beim Vergleich dieser Ausgabe mit der Ausgabe in CBASIC werden Sie feststellen, daß hier ein erheblicher Unterschied aufgetreten ist. Nach dem Verlassen der FUNCTION wird nicht nur der Variablen MULTI ein Wert zugewiesen, sondern es werden auch die für die lokalen Variablen a,b,c berechneten Ausdrücke an das Hauptprogramm übergeben und schließlich über die WRITE-Anweisung ausgegeben. Die Variablen a,b und c heißen deshalb lokal, weil sie nur in der FUNCTION MULTI lokale Gültigkeit besitzen. Es wäre also auch möglich, im Hauptprogramm mit genau den gleichen Variablennamen a,b und c zu arbeiten, die dann beliebige andere Werte annehmen könnten. Beim Einsprung in die FUNCTION wird der Inhalt der Variablen e,f und g entsprechend ihrer Reihenfolge in der Parameterliste an die Variablen a,b,c übergeben, wobei der eigentliche Variablenname überhaupt keine Rolle spielt. In der FUNCTION wird also gerechnet mit a=2,b=3 und c=4. Noch scheint alles genauso abzulaufen, wie unter CBASIC. Jetzt kommt aber der entscheidende Unterschied. Beim Erreichen der RETURN-Anweisung werden die neu berechneten Werte für a,b und c wieder entsprechend der Parameterliste an die Variablen e,f und g im Hauptprogramm zurückgegeben und sind von nun an nicht mehr definiert. Diese Übergabe ist ein typisches Beispiel für ein sogenanntes "call by reference". Beim Aufruf einer FUNCTION in FORTRAN wird also mit dem Inhalt der Variablen nicht der Variablenname, sondern die Referenz, d.h. die Speicheradresse der jeweiligen Variablen übergeben, was dazu führt, daß a und e,b und f,c und g den gleichen Speicherplatz belegen. Anders bei einem "call by value", das wir unter CBASIC kennengelernt haben. Hier wird eine Kopie der betreffenden Variablen erzeugt, die dann in der FUNCTION weiterverarbeitet wird. Nach dem Rücksprung in das Hauptprogramm ist das Original also noch vorhanden. Das spart natürlich nicht gerade Speicherplatz, kann aber bei der Rekursion unter PASCAL, also der Möglichkeit, daß eine FUNCTION sich selbst aufruft, sehr nützlich sein. Diese Möglichkeit bieten weder FORTRAN, das alle Variablen ausschließlich mit einem "call by reference" übergibt, noch BASIC. PASCAL sieht zusätzlich auch die Möglichkeit vor, Variablen mit einem "call by value" zu übergeben.

SUBROUTINES unter FORTRAN

Alles, was Sie bisher über FUNCTIONS erfahren haben, gilt im besonderen Maße auch für eine FORTRAN SUBROUTINE, die in der Konzeption noch stärker auf den Datentransfer mit einem "call by reference" ausgelegt ist. Eine FUNCTION muß bei der Rückkehr in das Hauptprogramm einen definierten Wert haben, d.h. in unserem Fall muß der Name MULTI mindestens einmal auf der linken Seite einer Gleichung in der FUNCTION auftauchen. Der eigentliche Sinn einer FUNCTION besteht deshalb nur darin, diesen einen Wert zu berechnen und nicht aus dem Abfallprodukt auch noch Variablen an das aufrufende Programm zurückzugeben. Wir haben diesen Weg jedoch gewählt, um einen Vergleich mit CBASIC zu ermöglichen. Wie ja bereits beim Test von CBASIC festgestellt werden mußte, ist das SUBROUTINE-Pendant GOSUB unter BASIC bedauerlicherweise nicht mehr, als ein etwas komfortableres GOTO.

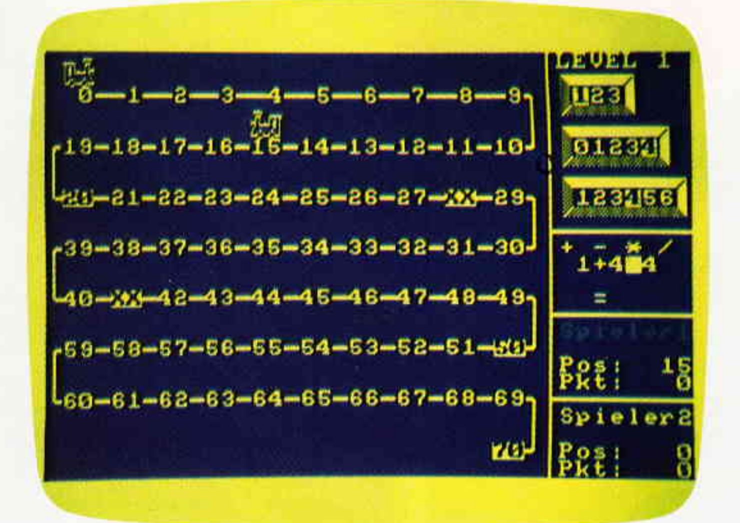
Damit ist das FORTRAN-Listing abgehandelt, wobei wir Ihnen natürlich nur einen kleinen Einblick in die Möglichkeiten von FORTRAN IV geben konnten. Wer noch mehr über FORTRAN erfahren möchte, sollte das NEVADA FORTRAN Handbuch zu Rate ziehen. Beim nächsten Mal geht es dann weiter mit PASCAL, wobei wir auf das vorgestellte FORTRAN-Listing noch einmal Bezug nehmen werden.

Dipl. Ing. H. Scheruhn


G·O·L·F
CONSTRUCTION SET
Nervenkitzel Loch und Löcher.

Lokomotive

Eine unterhaltsame Übung der vier Grundrechenarten für alle Lernenden und alle geneigten "Hirnakrobaten".



Viele Leser werden sicher noch den erleichterten Seufzer aus Ihrer Schulzeit im Ohr haben, mit dem die lästigen Hausaufgaben abgeschlossen wurden, um sich danach schleunigst der angenehmen Seite des "Schülerdaseins" zuzuwenden: dem Spielen. Und so mancher Leser wird diesen Seufzer mittlerweile von seinen eigenen Kindern zu hören bekommen: Hausaufgaben machen ist öde, viel lieber möchte man spielen! Da ersteres aber nun einmal nicht gerade unwichtig ist – wie vor allem Eltern und Lehrer den betroffenen Schüler immer wieder zu überzeugen versuchen, letzteres aber wesentlich mehr Vergnügen bereitet, bietet sich eine Verbindung von beidem ideal an. Ein Ergebnis dieser Bemühung haben Sie hier vor sich liegen.

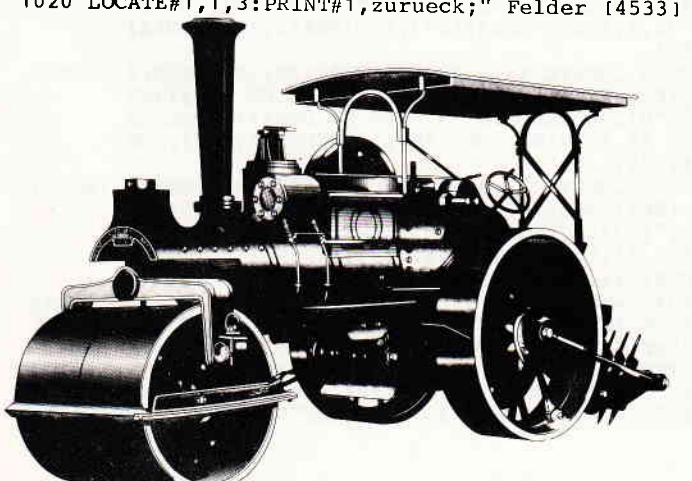
Zweck dieses für zwei Spieler ausgelegten Programmes ist es, wie aus dem Untertitel ja schon zu entnehmen ist, das Üben der vier Grundrechenarten. Dazu muß jeder Spieler seine Lokomotive so schnell wie möglich über eine (mit Haltepunkten versehene) Strecke zum Zielbahnhof fahren: Der Computer gibt drei Zahlen vor, und der Spieler muß diese mit Hilfe der Grundrechenarten derart verknüpfen, daß das Ergebnis ein größtmögliches Vorankommen ermöglicht. Dabei ist erstens auf die Regel "Punktrechnung vor Strichrechnung" zu achten, und zweitens muß die Rechnung eine ganze Zahl ergeben. Damit man nun die Operatoren nicht einfach auf gut Glück zwischen die gegebenen Zahlen setzt und sich vom Ergebnis überraschen läßt, will der Rechner dieses auch vom Spieler wissen. Hat man richtig gerechnet, so fährt die Lok des Spielers entsprechend des Ergebnisses vor oder – bei negativem Ergebnis – zurück. Allerdings liegen auf der Strecke Haltepunkte, die nicht überfahren werden dürfen; man muß sie genau erreichen und notfalls ein wenig "rangieren". Als Gegengewicht dazu darf man bei genauem Treffen eines Bonusfeldes zehn Felder vorrücken. Auch kann man seinen Gegenspieler ein wenig "ärgern": Gelingt der Versuch, sich auf das gleiche Feld wie dieser zu setzen, so muß er 20 Felder zurückfahren.

Na denn – viel Spaß beim Pauken! (V.Reher/MC)

für 464-664-6128

```
10 REM ********************************* [1586]
20 REM *           Lokomotive          * [755]
30 REM *              von              * [556]
40 REM *           V.Reher        4.86 * [1348]
50 REM ********************************* [1586]
60 MODE 1:INK 0,1:INK 1,24:INK 2,1,24:INK [5397]
3,12,24:PAPER 0:PEN 1:BORDER 1:SPEED INK 6
,20
70 DEFINT a-y:DEFREAL s                   [1247]
80 DIM strecke(70):posi(1)=0:posi(2)=0:str [3121]
ecke(70)=1
90 zurueck=20:REM gegnerische Lok zuruecks [2318]
etzen
100 rz$="+-*/"                            [540]
110 RANDOMIZE TIME                        [1777]
120 GOSUB 2370                            [817]
130 WINDOW#1,1,25,23,25                   [1079]
140 REM ********** Titelbild ***********  [1815]
150 x$="LOKOMOTIVE "                      [576]
160 FOR x=34 TO 1 STEP-1:LOCATE x,12:PRINT [6956]
 lokl$:SOUND 7,5,10,,,,20:FOR p=1 TO 200:N
EXT:LOCATE x,12:PRINT loesch$
170 TAG:MOVE 620-(40-x)*10,206:PRINT x$;:T [2100]
AGOFF
180 NEXT                                  [350]
190 ENV 1,1,15,1,100,0,1,12,-1,8,2,-1,20  [2238]
200 SOUND 7,880,200,0,1                   [1484]
210 FOR x=24 TO 1 STEP-1:INK 1,x:FOR p=1 T [3240]
O 200:NEXT:NEXT
220 CLS:INK 1,24                          [817]
230 REM ******** Namen eingeben *******   [1902]
240 LOCATE 1,3:PRINT"Spieler 1 - Bitte dei [4498]
nen Namen eingeben"
250 LOCATE 1,5:PRINT">":xp=3:yp=5:lang=15: [4014]
lasc=32:hasc=122:GOSUB 1870
260 name1$=UPPER$(LEFT$(tx$,1))+LOWER$(MID [3035]
$(tx$,2))
270 LOCATE 1,7:PRINT"Spieler 2 - Bitte dei [5242]
nen Namen eingeben"
280 LOCATE 1,9:PRINT">":yp=9:GOSUB 1870   [1457]
290 name2$=UPPER$(LEFT$(tx$,1))+LOWER$(MID [2995]
$(tx$,2))
300 PRINT:PRINT:PRINT"Braucht ihr die Spie [3463]
lregeln";
310 PRINT" (j/n)";:PEN 2:PRINT" ?";:PEN 1 [2480]
320 xp=37:yp=VPOS(#0):lang=1:lasc=74:hasc= [4815]
110:GOSUB 1870
330 IF LOWER$(tx$)="n" THEN 490           [1620]
340 IF LOWER$(tx$)<>"j" THEN 320          [1969]
350 REM ******** Spielregeln **********   [1988]
360 CLS:PRINT:PRINT"Eure Aufgabe besteht d [8937]
arin, eine":PRINT"Lokomotive ueber die Sch
ienen bis auf"
370 PRINT"die Zahl 70 zu fahren.":PRINT   [3652]
380 PRINT"Die Lokomotive bewegt sich, wenn [6435]
 ihr":PRINT"die Rechenzeichen '+-*/' zwisc
hen die"
390 PRINT"drei zufaellig gezogenen Zahlen [5150]
setzt.":PRINT
400 PRINT"Die "CHR$(24)"invers"CHR$(24) " [10571]
dargestellten Zahlen muessen":PRINT"genau
erreicht werden."
410 PRINT"Ist das Ergebnis der Rechnung ke [12257]
ine":PRINT"ganze Zahl, dann bleibt die Lok
 stehen":PRINT"und der andere Spieler ist
an der":PRINT"Reihe."
420 PRINT"Ist die Zahl negativ, dann faehr [4316]
t die":PRINT"Lok zurueck."
430 PRINT"Wenn eine Lok auf das "CHR$(24)" [10306]
XX"CHR$(24)" kommt, dann":PRINT"faehrt sie
 automatisch 10 Felder in":PRINT"Richtung
Ziel."
440 PRINT:PRINT"Kommt eine Lok auf ein Fel [8117]
d, auf dem":PRINT"bereits die andere Lok s
teht, so muss":PRINT"die letztere 20 Felde
r zurueck."
450 PEN 3:LOCATE 6,25:PRINT"> Starten mit [4590]
SPACE-Taste <":PEN 1
460 x$=INKEY$:IF x$="" THEN 460           [2640]
470 IF ASC(x$)<>32 THEN 460               [1213]
480 REM *********** Level ************     [1330]
490 FOR x=0 TO 69:strecke(x)=0:NEXT       [1860]
500 CLS:BORDER 1:PRINT:PRINT"Welcher Schwi [4962]
erigkeitsgrad ?"
510 PRINT:PRINT:PRINT"       1 = leicht":P [6398]
RINT:PRINT"       2 = mittel":PRINT:PRINT"
       3 = schwer"
520 x$=INKEY$:IF x$="" THEN 520           [2676]
530 x=ASC(x$):IF x<49 OR x>51 THEN 520    [1826]
540 IF x=49 THEN level=1:strecke(20)=1:str [5015]
ecke(28)=2:strecke(41)=2:strecke(50)=1
550 IF x=50 THEN level=2:strecke(20)=1:str [5965]
ecke(26)=2:strecke(35)=1:strecke(38)=2:str
ecke(50)=1:strecke(60)=1
560 IF x=51 THEN level=3:strecke(10)=1:str [5344]
```

```
ecke(20)=1:strecke(40)=1:strecke(23)=2:str
ecke(50)=1:strecke(60)=1
570 BORDER 12:INK 2,24:INK 3,24              [1708]
580 REM ******** spieler 1 **********         [1746]
590 CLS:GOSUB 1980                           [904]
600 LOCATE 2,1:PRINT lokr$                   [1485]
610 GOSUB 2300                               [863]
620 INK 2,1,24:INK 3,24                      [786]
630 GOSUB 1360                               [923]
640 GOSUB 1560                               [899]
650 GOSUB 1690:IF dezimal THEN dezimal=0:G   [4030]
OSUB 960:GOTO 780
660 IF falsch THEN falsch=0:GOSUB 990:GOTO   [2610]
 780
670 FOR x=posi(1)+1 TO posi(1)+summe-1       [2951]
680 IF strecke(x)=1 THEN merker=-1:GOSUB 1   [3018]
050:x=posi(1)+summe
690 NEXT x                                   [356]
700 IF merker THEN merker=0:GOTO 780         [1988]
710 IF SGN(summe)<0 THEN st=-1:summe=-MIN(   [4199]
ABS(posi(1)),ABS(summe)) ELSE st=1
720 y=posi(1):posi(1)=posi(1)+summe:GOSUB    [4281]
1120:summe=0:y=posi(2):GOSUB 1120
730 LOCATE 37,19:PRINT USING"####";posi(1)   [3785]
740 IF posi(1)=70 THEN 1200                  [1521]
750 IF strecke(posi(1))=2 THEN GOSUB 1080:   [3096]
summe=10:GOTO 710
760 IF posi(1)<>0 THEN IF posi(1)=posi(2)    [12027]
THEN GOSUB 1020:summe=-MIN(posi(1),zurueck
):y=posi(2):posi(2)=posi(2)+summe:st=-1:GO
SUB 1120:summe=0:y=posi(1):GOSUB 1120:LOCA
TE 37,24:PRINT USING"####";posi(2)
770 REM ********* spieler 2 *********         [1357]
780 INK 2,24:INK 3,1,24                      [271]
790 GOSUB 2300                               [863]
800 GOSUB 1360                               [923]
810 GOSUB 1560                               [899]
820 GOSUB 1690:IF dezimal THEN dezimal=0:G   [4056]
OSUB 960:GOTO 610
830 IF falsch THEN falsch=0:GOSUB 990:GOTO   [2692]
 610
840 FOR x=posi(2)+1 TO posi(2)+summe-1       [3105]
850 IF strecke(x)=1 THEN merker=-1:GOSUB 1   [3627]
050:x=posi(2)+summe
860 NEXT x                                   [356]
870 IF merker THEN merker=0:GOTO 610         [1998]
880 IF SGN(summe)<0 THEN st=-1:summe=-MIN(   [3571]
ABS(posi(2)),ABS(summe)) ELSE st=1
890 y=posi(2):posi(2)=posi(2)+summe:GOSUB    [5322]
1120:summe=0:y=posi(1):GOSUB 1120
900 LOCATE 37,24:PRINT USING"####";posi(2)   [2127]
910 IF posi(2)=70 THEN 1200                  [1985]
920 IF strecke(posi(2))=2 THEN GOSUB 1080:   [2301]
summe=10:GOTO 880
930 IF posi(2)<>0 THEN IF posi(2)=posi(1)    [14193]
THEN GOSUB 1020:summe=-MIN(posi(2),zurueck
):y=posi(1):posi(1)=posi(1)+summe:st=-1:GO
SUB 1120:summe=0:y=posi(2):GOSUB 1120:LOCA
TE 37,19:PRINT USING"####";posi(1)
940 GOTO 610                                 [411]
950 REM ********** Meldungen ********         [2141]
960 LOCATE#1,1,3:PRINT#1," Das Ergebnis is   [7829]
t nicht":PRINT#1," ganzzahlig !":PRINT#1,"
 Ergebnis=";:PRINT#1,USING"##.###";summe;
970 FOR p=1 TO 5000:NEXT                     [1354]
980 CLS#1:RETURN                             [1242]
990 LOCATE#1,1,3:PRINT#1," Das Ergebnis is   [11333]
t leider":PRINT#1," nicht richtig !":PRINT
#1," Ergebnis=";:PRINT#1,USING"##";summe;
1000 FOR p=1 TO 5000:NEXT                    [1354]
1010 CLS#1:RETURN                            [1242]
1020 LOCATE#1,1,3:PRINT#1,zurueck;" Felder   [4533]
```



BEKANNTMACHUNG

Bei unserem allseits bekannten und beliebten Telefon-Service, dem „Heißen Draht", gibt es eine Veränderung. Sie können Ihre Fragen und Anregungen von

17^{00} – 20^{00} Uhr

an die Redaktion von Schneider CPC International richten. Auf Ihren Anruf freuen sich: Michael Ceol (Joyce), Michael Ebbrecht (Hardware), Stefan Ritter (Redaktion), Thomas Morgen (Programmierung) und Heinrich Stiller (Spiele/Adventures).

JEDEN MITTWOCH AM

HEISSEN DRAHT

Tel.: 05651/8702

```
 zurueck !"
1030 FOR p=1 TO 6000:NEXT                [1184]
1040 CLS#1:RETURN                        [1242]
1050 LOCATE#1,1,3:PRINT#1," Du willst doch [7031]
 nicht die";:PRINT#1," naechste Haltestell
e":PRINT#1," ueberfahren ?"
1060 FOR p=1 TO 6000:NEXT                [1184]
1070 CLS#1:RETURN                        [1242]
1080 LOCATE#1,1,3:PRINT#1," 10 Felder vorr [3147]
uecken !"
1090 FOR p=1 TO 5000:NEXT                [1354]
1100 CLS#1:RETURN                        [1242]
1110 REM ********* Lok bewegen ********  [1454]
1120 lok$=lokr$:FOR x=y TO y+summe STEP st [2982]
1130 IF (x>9 AND x<20) OR (x>29 AND x<40) [8805]
OR (x>49 AND x<60) OR x>69  THEN xp=29-VAL
(RIGHT$(STR$(x),1))*3:lok$=lokl$:GOTO 1150
1140 xp=VAL(RIGHT$(STR$(x),1))*3+2:lok$=lo [3110]
kr$
1150 yp=INT(x/10)*3+1:LOCATE xp,yp:PRINT l [7880]
ok$:SOUND 7,50,10,,,,20:FOR p=1 TO 200:NEX
T:LOCATE xp,yp:PRINT loesch$
1160 NEXT                                [350]
1170 LOCATE xp,yp:PRINT lok$             [2507]
1180 RETURN                              [555]
1190 REM ********* Spiel-Ende *********  [1670]
1200 punkte1=punkte1+posi(1)             [2142]
1210 punkte2=punkte2+posi(2)             [2338]
1220 LOCATE 37,20:PRINT USING"####";punkte [1369]
1
1230 LOCATE 37,25:PRINT USING"####";punkte [1917]
2;
1240 INK 2,24:INK 3,24                   [183]
1250 IF punkte1>punkte2 THEN LOCATE#1,1,3: [5095]
PRINT#1," Damit hast du gewonnen":PRINT#1,
" ";name1$
1260 IF punkte1<punkte2 THEN LOCATE#1,1,3: [6655]
PRINT#1," Damit hast du gewonnen":PRINT#1,
" ";name2$
1270 FOR p=1 TO 8000:NEXT                [809]
1280 CLS#1                               [373]
1290 PRINT#1,"1 = Spiel fortsetzen":PRINT# [5004]
1,"2 = Spiel neu starten":PRINT#1,"3 = End
e";
1300 x$=INKEY$:IF x$="" THEN 1300        [2596]
1310 x=ASC(x$):IF x<49 OR x>51 THEN 1300 [1713]
1320 IF x=50 THEN posi(1)=0:posi(2)=0:GOTO [1620]
 220
1330 IF x=51 THEN CLS:PRINT"Das Programm k [5849]
ann mit":PRINT"CONT <RETURN>":PRINT"fortge
setzt werden":END
1340 posi(1)=0:posi(2)=0:GOTO 590        [1101]
1350 REM ******** Zufallszahlen *******   [2070]
1360 zahl(1)=INT(RND*3)+1:zahl(2)=INT(RND* [4998]
5):zahl(3)=INT(RND*6)+1:lauf=12:z=lauf-5
1370 FOR y=1 TO lauf                     [1305]
1380 FOR x=1 TO 3:SOUND 7,100,1,7        [2006]
1390 PRINT CHR$(24):LOCATE 33+x,3:PRINT RI [8825]
GHT$(STR$(x),1):PRINT CHR$(24):LOCATE 33+x
,3:FOR p=1 TO y*12:NEXT p:PRINT RIGHT$(STR
$(x),1):IF y>z AND x=zahl(1) THEN x=3:y=la
uf
1400 NEXT x:NEXT y                       [980]
1410 PRINT CHR$(24):LOCATE 33+zahl(1),3:PR [4113]
INT RIGHT$(STR$(zahl(1)),1):PRINT CHR$(24)
1420 FOR y=1 TO lauf                     [1305]
1430 FOR x=0 TO 4:SOUND 7,150,1,7        [2083]
1440 PRINT CHR$(24):LOCATE 34+x,6:PRINT RI [11765]
GHT$(STR$(x),1):PRINT CHR$(24):LOCATE 34+x
,6:FOR p=1 TO y*8:NEXT p:PRINT RIGHT$(STR$
(x),1):IF y>z AND x=zahl(2) THEN x=4:y=lau
f
1450 NEXT x:NEXT y                       [980]
1460 PRINT CHR$(24):LOCATE 34+zahl(2),6:PR [4241]
INT RIGHT$(STR$(zahl(2)),1):PRINT CHR$(24)
1470 FOR y=1 TO lauf                     [1305]
1480 FOR x=1 TO 6:SOUND 7,200,1,7        [2069]
1490 PRINT CHR$(24):LOCATE 33+x,9:PRINT RI [8306]
GHT$(STR$(x),1):PRINT CHR$(24):LOCATE 33+x
,9:FOR p=1 TO y*6:NEXT p:PRINT RIGHT$(STR$
(x),1):IF y>z AND x=zahl(3) THEN x=6:y=lau
f
1500 NEXT x:NEXT y                       [980]
1510 PRINT CHR$(24):LOCATE 33+zahl(3),9:PR [3143]
INT RIGHT$(STR$(zahl(3)),1):PRINT CHR$(24)
1520 LOCATE 33,13:PRINT zahl(1):LOCATE 35, [6002]
13:PRINT zahl(2):LOCATE 37,13:PRINT zahl(3
)
1530 LOCATE 35,15:PRINT"="               [1902]
```

```
1540 RETURN                              [555]
1550 REM **** Eingabe Rechenzeichen ***  [2349]
1560 xp=35:yp=13:lasc=42:hasc=47:lang=1:tx [3400]
$=" ":GOSUB 1870
1570 IF INSTR(rz$,tx$)=0 THEN 1560       [2037]
1580 IF zahl(2)=0 AND tx$="/" THEN 1560  [1612]
1590 rz1$=tx$                            [916]
1600 xp=37:GOSUB 1870                    [2271]
1610 IF INSTR(rz$,tx$)=0 THEN 1560       [2037]
1620 rz2$=tx$                            [1012]
1630 REM ****** Eingabe Ergebnis ******  [1608]
1640 xp=36:yp=15:lasc=45:hasc=57:lang=2:GO [4069]
SUB 1870
1650 IF INSTR(tx$,"/")<>0 THEN 1640      [1454]
1660 IF INSTR(tx$,".")<>0 THEN 1640      [1786]
1670 ergebnis=VAL(tx$)                   [1940]
1680 REM ********* Berechnung *********  [1924]
1690 IF rz2$="/" AND rz1$="/" THEN summe=z [2895]
ahl(1)/zahl(2)/zahl(3):GOTO 1830
1700 IF rz2$="/" AND rz1$<>"*" THEN zwsumm [2722]
e=zahl(2)/zahl(3):GOTO 1800
1710 IF rz2$="*" AND rz1$="/" THEN summe=z [5243]
ahl(1)/zahl(2)*zahl(3):GOTO 1830
1720 IF rz2$="*" THEN zwsumme=zahl(2)*zahl [1984]
(3):GOTO 1800
1730 IF rz1$="+" THEN zwsumme=zahl(1)+zahl [3754]
(2):GOTO 1770
1740 IF rz1$="-" THEN zwsumme=zahl(1)-zahl [3443]
(2):GOTO 1770
1750 IF rz1$="*" THEN zwsumme=zahl(1)*zahl [1719]
(2):GOTO 1770
1760 zwsumme=zahl(1)/zahl(2)             [2390]
1770 IF rz2$="+" THEN summe=zwsumme+zahl(3 [2447]
):GOTO 1830
1780 IF rz2$="-" THEN summe=zwsumme-zahl(3 [3998]
):GOTO 1830
1790 summe=zwsumme/zahl(3):GOTO 1830     [2680]
1800 IF rz1$="+" THEN summe=zahl(1)+zwsumm [2550]
e:GOTO 1830
1810 IF rz1$="-" THEN summe=zahl(1)-zwsumm [2927]
e:GOTO 1830
1820 summe=zahl(1)*zwsumme:GOTO 1830     [1617]
1830 IF INT(summe)<>summe THEN dezimal=-1: [2036]
GOTO 1850
1840 IF summe<>ergebnis THEN falsch=-1   [2521]
1850 RETURN                              [555]
1860 REM ******* Eingaberoutine *******  [779]
1870 WHILE INKEY$<>"":WEND:LOCATE xp,yp:PR [5208]
INT SPACE$(LEN(tx$)):tx$="":LOCATE xp,yp:C
ALL &BB81
1880 x$=INKEY$:IF x$="" THEN 1880        [2672]
1890 x=ASC(x$):IF x=13 AND tx$="" THEN 196 [2872]
0
1900 IF x=127 AND LEN(tx$)=0 THEN GOTO 196 [1546]
0
1910 IF x=13 THEN CALL &BB84:RETURN      [1578]
1920 IF x=127 THEN tx$=LEFT$(tx$,LEN(tx$)- [3162]
1):PRINT CHR$(8);" ";CHR$(8);:GOTO 1880
1930 IF LEN(tx$)=lang THEN 1960          [1294]
1940 IF x<lasc OR x>hasc THEN 1960       [2079]
1950 tx$=tx$+x$:PRINT x$;:GOTO 1880      [407]
1960 SOUND 4,1000,5,7,,,15:GOTO 1880     [2266]
1970 REM ******* Bildschirmmaske ******  [1842]
1980 BORDER 24:FOR y=1 TO 25:LOCATE 32,y:P [3623]
RINT CHR$(149);:NEXT
1990 FOR y=11 TO 22 STEP 5:LOCATE 32,y:PRI [4708]
NT CHR$(151);STRING$(8,154):NEXT
2000 LOCATE 33,1:PRINT"LEVEL";level      [2207]
2010 LOCATE 33,2:PRINT CHR$(185);STRING$(3 [10906]
,186);CHR$(187):LOCATE 33,3:PRINT CHR$(188
);"123":LOCATE 37,3:PRINT CHR$(192):LOCATE
 33,4:PRINT CHR$(189);STRING$(3,190);CHR$(
191)
2020 LOCATE 33,5:PRINT CHR$(185);STRING$(5 [11868]
,186);CHR$(187):LOCATE 33,6:PRINT CHR$(188
);"01234":LOCATE 39,6:PRINT CHR$(192):LOCA
TE 33,7:PRINT CHR$(189);STRING$(5,190);CHR
$(191)
2030 LOCATE 33,8:PRINT CHR$(185);STRING$(6 [8653]
,186);CHR$(187):LOCATE 33,9:PRINT CHR$(188
);"123456":LOCATE 40,9:PRINT CHR$(192):LOC
ATE 33,10:PRINT CHR$(189);STRING$(6,190);C
HR$(191)
2040 LOCATE 33,12:PRINT"+ - * /"         [1363]
2050 INK 2,24:PEN 2:LOCATE 33,17:PRINT"Spi [10176]
eler1";:PEN 1:LOCATE 33,19:PRINT "Pos:";:P
RINT USING"####";posi(1):LOCATE 33,20:PRIN
T "Pkt:";:PRINT USING"####";punkte1
2060 PEN 3:LOCATE 33,22:PRINT"Spieler2";:P [8853]
```

```
EN 1:LOCATE 33,24:PRINT"Pos:";:PRINT USING
"####";posi(1):LOCATE 33,25:PRINT "Pkt:";:
PRINT USING"####";punkte2;
2070 LOCATE 3,3:PRINT"0";:FOR x=1 TO 9:PRI  [5711]
NT CHR$(154)CHR$(154)RIGHT$(STR$(x),1);:NE
XT:PRINT vbr$
2080 LOCATE 1,9:PRINT vbl$;:PRINT"19";:FOR  [5892]
 x=18 TO 10 STEP-1:PRINT CHR$(154)RIGHT$(S
TR$(x),2);:NEXT
2090 LOCATE 2,9:PRINT"20";:FOR x=21 TO 29:  [5134]
PRINT CHR$(154)RIGHT$(STR$(x),2);:NEXT:PRI
NT vbr$
2100 LOCATE 1,15:PRINT vbl$;:PRINT"39";:FO  [5051]
R x=38 TO 30 STEP-1:PRINT CHR$(154)RIGHT$(
STR$(x),2);:NEXT
2110 LOCATE 2,15:PRINT"40";:FOR x=41 TO 49  [6241]
:PRINT CHR$(154)RIGHT$(STR$(x),2);:NEXT:PR
INT vbr$
2120 LOCATE 1,21:PRINT vbl$;:PRINT"59";:FO  [5801]
R x=58 TO 50 STEP-1:PRINT CHR$(154)RIGHT$(
STR$(x),2);:NEXT
2130 LOCATE 2,21:PRINT"60";:FOR x=61 TO 69  [5945]
:PRINT CHR$(154)RIGHT$(STR$(x),2);:NEXT:PR
INT vbr$
2140 LOCATE 29,24:PRINT "70"                [1796]
2150 REM ****** Bahnhoefe markieren ***     [1592]
2160 fl=0:xp=29:yp=6:PRINT CHR$(24)         [1433]
2170 FOR x=10 TO 70                         [1034]
2180 IF fl THEN 2230                        [670]
2190 LOCATE xp,yp:IF strecke(x)=1 THEN PRI  [4245]
NT USING"##";x;
2200 LOCATE xp,yp:IF strecke(x)=2 THEN PRI  [3445]
NT "XX";
2210 IF RIGHT$(STR$(x),1)="9" THEN fl=NOT   [3985]
fl:yp=yp+3:GOTO 2270
2220 xp=xp-3:GOTO 2270                      [1324]
2230 LOCATE xp,yp:IF strecke(x)=1 THEN PRI  [4245]
NT USING"##";x;
2240 LOCATE xp,yp:IF strecke(x)=2 THEN PRI  [3445]
NT "XX";
2250 IF RIGHT$(STR$(x),1)="9" THEN fl=NOT   [3985]
fl:yp=yp+3:GOTO 2270
2260 xp=xp+3                                [1152]
2270 NEXT:PRINT CHR$(24)                    [1373]
2280 RETURN                                 [555]
2290 REM ****** Anzeige loeschen ******     [1793]
2300 LOCATE 34,3:PRINT"123"                 [1028]
2310 LOCATE 34,6:PRINT"01234"               [1516]
2320 LOCATE 34,9:PRINT"123456"              [1431]
2330 LOCATE 34,13:PRINT"      "             [1410]
2340 LOCATE 35,15:PRINT"     "              [1909]
2350 RETURN                                 [555]
2360 REM ***** Zeichen definieren *****     [2172]
2370 SYMBOL AFTER 175                       [1433]
2380 REM ****** Lokomotive rechts *****     [1826]
2390 SYMBOL 176,0,0,0,0,0,254,66,90         [1308]
2400 SYMBOL 177,0,0,0,62,34,28,20,21        [1907]
2410 SYMBOL 178,90,91,88,64,76,82,115,12    [2092]
2420 SYMBOL 179,151,245,4,2,50,73,207,48    [2055]
2430 REM ****** Lokomotive links ******     [1796]
2440 SYMBOL 180,0,0,0,124,68,56,40,168      [2172]
2450 SYMBOL 181,0,0,0,0,0,127,66,90         [1978]
2460 SYMBOL 182,233,175,32,64,76,146,243,1  [2473]
2
2470 SYMBOL 183,90,218,26,2,50,74,206,48    [2123]
2480 REM ** Rahmen fuer Zufallszahlen *     [1045]
2490 SYMBOL 185,0,0,63,31,47,23,43,20       [1850]
2500 SYMBOL 186,0,0,255,255,255,255,255,0   [2453]
2510 SYMBOL 187,0,0,252,244,236,220,188,12  [2518]
4
2520 SYMBOL 188,42,20,42,20,42,20,42,20     [1871]
2530 SYMBOL 189,43,22,45,26,53,0,0,0        [1870]
2540 SYMBOL 190,85,170,85,170,85,0,0,0      [2488]
2550 SYMBOL 191,124,188,92,172,84,0,0,0     [2336]
2560 SYMBOL 192,124,124,124,124,124,124,12  [2255]
4,124
2570 lokr$=CHR$(176)+CHR$(177)+CHR$(8)+CHR  [3125]
$(8)+CHR$(10)+CHR$(178)+CHR$(179)
2580 lokl$=CHR$(180)+CHR$(181)+CHR$(8)+CHR  [4076]
$(8)+CHR$(10)+CHR$(182)+CHR$(183)
2590 loesch$=CHR$(32)+CHR$(32)+CHR$(8)+CHR  [3515]
$(8)+CHR$(10)+CHR$(32)+CHR$(32)
2600 vbr$=CHR$(156)+CHR$(8)+CHR$(10)+CHR$(  [6417]
149)+CHR$(8)+CHR$(10)+CHR$(149)+CHR$(8)+CH
R$(10)+CHR$(153)
2610 vbl$=CHR$(147)+CHR$(8)+CHR$(11)+CHR$(  [4801]
149)+CHR$(8)+CHR$(11)+CHR$(149)+CHR$(8)+CH
R$(11)+CHR$(150)
2620 RETURN                                 [555]
```

Korrektur zu Buchbesprechung "CP/M Plus" von Jürgen Hückstädt

Bei der Besprechung des "CP/M Plus"-Buches vom Markt & Technik-Verlag (Heft 6/86, S. 124) ist uns leider ein Fehler unterlaufen. Der letzte Absatz (ab: „Die Krönung ist...") gehört zu einem anderen Buch, Jürgen Hückstädt's sehr gutes Buch enthält keinen Anhang.
Dieses Versehen bitten wir vielmals zu entschuldigen.

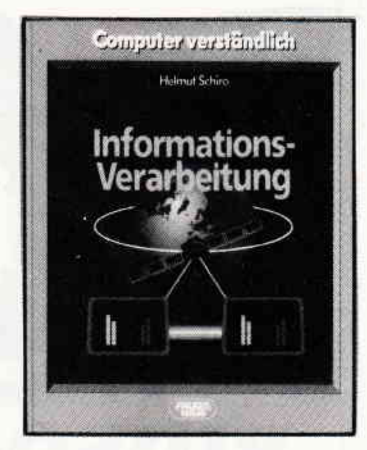
Mühleisen/Seipel
Vom Problem zum Algorithmus
Mentor-Verlag 1986
144 Seiten
ISBN 3-580-64750-4
Preis: 21,80 DM

Der Mentor Verlag (der übrigens zur Langenscheidt Verlagsgruppe gehört) stellt hier den ersten Band aus der Reihe "Informatik-Grundlagen für die systematische Programmierung von Computern" vor. Im Mittelpunkt dieses Bandes steht hierbei die Umwandlung des Problems zu einem Algorithmus.
Anhand zahlreicher und gut erläuterten Beispiele wird der Lernende Schritt für Schritt in die Materie eingeführt. Am Anfang steht das Erkennen und Umsetzen eines Problems in einen Programmablauf. Dabei werden die Grundregeln eingehalten, so daß die erworbenen Kenntnisse auf jede beliebige Programmiersprache angewandt werden können.

Ein interessantes und empfehlenswertes Buch, das sich nicht nur an Schüler wendet.

Schiro
Grundwissen Informationsverarbeitung
Falken Verlag 1986
311 Seiten
ISBN 3-8068-4314-7
Preis: 58,– DM

Am Anfang war der Abakus. Mit dem vorliegenden Buch gibt der Autor einen umfassenden Überblick über Entwicklung, Geschichte und Anwendungen der Computertechnologie.
Die Thematik umfaßt sowohl die Entwicklung von Hard- und Software, als auch die neuen Kommunikationsmedien.
Der Autor beschreibt in leicht verständlicher Form das Zusammenspiel von Mensch und Maschine. Selbst komplizierte Sachverhalte werden anschaulich erläutert.
Aufgrund der gegebenen Thematik richtet sich dieses Buch vor allem an diejenigen, die als Laien mit dieser Technik konfrontiert werden, ist aber auch als Nachschlagewerk für Schule und Studium sehr gut geeignet.



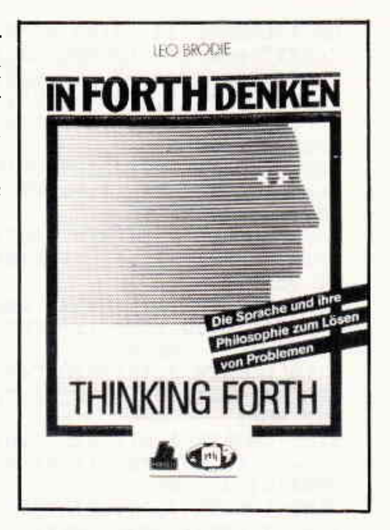
Hausbacher
Das Prozessor-Buch zum Z-80
Data Becker 1985
563 Seiten
ISBN 3-89011-096-7
Preis: 59,– DM

Das ideale Buch für Techniker und Maschinenspracheprogrammierer.
Hauptbestandteil dieses sehr umfangreichen Werkes ist der Z-80-Befehlssatz, der sehr ausführlich beschrieben und mit Beispielen versehen ist.
Weitere Themen sind die serielle und parallele Datenübertragung, Ein-/Ausgabetechniken sowie die Anschlußmöglichkeiten von Systembausteinen.
Ein rundherum als gut gelungenes und effektiv einzusetzendes Lehr- und Lernbuch zu bezeichnendes Werk, dessen Einsatzmöglichkeiten kaum eingegrenzt werden können. Dieses Buch ist übrigens auf alle Rechner mit Z-80 CPU anzuwenden und gibt Technik und Programmierung des Z-80 in leicht verständlicher Form wieder.



Brodie
In Forth denken
Carl Hanser-Verlag 1986
285 Seiten
ISBN 3-446-14334-3
Preis: 48,– DM

Die Programmiersprache, bzw. das Betriebssystem FORTH findet heute in vielen Bereichen seine Anwendungen. Der Autor versucht hier, die Grundzüge von Forth in einfacher Form darzustellen und dem Lernenden die strukturierte Programmierung nahezubringen. So wird ein gewichtiger Teil auf die Softwareentwicklung gelegt, dieses Kapitel dürfte auf nahezu jede Programmiersprache angewendet werden.



Vorkenntnisse sind zwar nicht unbedingt notwendig, helfen in einigen Kapiteln jedoch über mögliche Verständnisschwierigkeiten hinweg, denn das Hauptaugenmerk liegt eindeutig auf dem Erlernen eines sicheren und effektiven Programmierstils in Forth.
Mit diesem Buch liegt wohl eines der interessantesten Forth-Arbeitsbücher vor und ist für Interessierte, Fortgeschrittene und Profis uneingeschränkt zu empfehlen.

Borgersen
Von Basic zu Pascal
Elektra Verlag 1984
138 Seiten
ISBN 3-922238-61-0
Preis: 34,– DM

In letzter Zeit kristallisiert sich die Programmiersprache Pascal immer mehr zu einem Standard für Mikrocomputer heraus. Zur Zeit gibt es mehrere Pascal-Versionen; u.a. auch das sogenannte USCD-Pascal, mit dem sich dieses Buch beschäftigt. Dem Basic-Programmierer wird hier eine Einführung in Pascal gegeben; die großen Unterschiede zu Basic werden deutlich. Dieses Buch versteht sich daher auch nicht als Pascal-Programmsammlung, sondern als Übergang vom Kauderwelsch Basic zum strukturierten Pascal. Als hilfreich erweist sich für den Pascal-Laien die im Anhang befindliche Referenzliste, die alle wesentlichen Schlüsselwörter vergleichend darstellt.

Das Tor zur Welt

öffnet sich für alle CPC-Besitzer mit einem von der Firma SIKOS angebotenen System von Peripheriekarten auf der Basis des bekannten ECB-Bussystems. Die Stärken dieses Busses wurden in der SCHNEIDERWARE-Serie gelegentlich erwähnt; die größten Vorteile bestehen in der großen Verbreitung des Systems und der Kompatibilität zu allen Z-80-Rechnern.

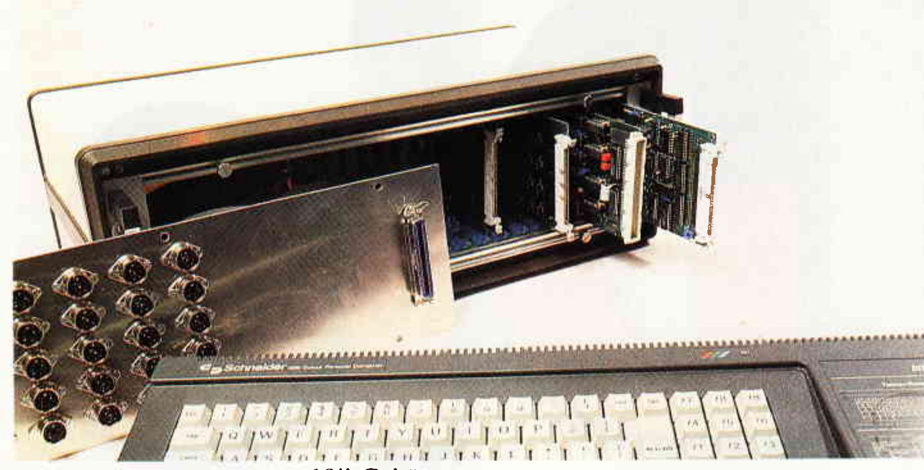


Bild 1: Der ECB-Bus im 19"-Gehäuse

Die von o.g. Hersteller angebotenen Karten zielen in Qualität und Anwendungsmöglichkeiten deutlich auf professionelle Nutzung; die Preisgestaltung der Bausteine ermöglicht aber durchaus dem interessierten Privatmann den Einsatz des Systems im trauten Heim.

1. Die Interfacekarte
Sie ist Voraussetzung für den Betrieb der CPC's am ECB-Bus; hier wird die Pinbelegung des CPC-Expansionports auf die Pins des ECB-Bus konvertiert; ebenso findet auf dieser Karte die Bufferung zum Schutz des CPC-Ports statt. Diese Karte enthält drei Steckplätze; je einen für das Kabel zum Anschluß von 464/664 bzw. 6128; der dritte erlaubt die Verbindung mit allerlei Peripherie wie Disk-Laufwerke oder Speichererweiterungen (wichtig für den 464). Ein besonderes Merkmal dieses Interfaces ist die Möglichkeit, den Bus in den Zweck-DMA-Betrieb umzuschalten oder einen Sub-Prozessor anzuschließen.

Die Optokoppler-Karte stellt 32 potentialfreie Optokoppler zur Verfügung, die für Schaltanwendungen eingesetzt werden können, die eine galvanische Trennung von Eingang und Ausgang erfordern. Die maximal mögliche Eingangsspannung beträgt 30 V; Signale werden ab 4 mA einwandfrei erkannt; die Stromaufnahme der Karte beträgt nur etwa 270 mA bei 5V. Defekte Optokoppler können problemlos einzeln ausgewechselt werden; im Servicefall ein großer Vorteil.

2. Die Relais-Karte
Sie enthält 16 potentialfreie Relaisausgänge (Öffner und Schließer); die Kontakte können mit Lötpillen auf High (5V)- oder Low (GND)-Potential verbunden werden. Die Schaltleistung der Relaiskontakte beträgt maximal 24 W bei einer maximalen Schaltspannung von 30 V Gleich-/150 V Wechselspannung und einem maximalen Schaltstrom von 1 A. Die Relaisadressen sind per DIL-Schalter einstellbar; ein RESET setzt alle Relais in Grundstellung zurück. Durch Ansprechen einer Portadresse können alle Relais rückgesetzt werden; dies entspricht einer softwaremäßigen Notaus-Funktion.
Die Wandlerkarte beherbergt in der höchsten Ausbaustufe maximal drei A/D-Wandler und einen D/A-Wandler. Die A/D-Wandler arbeiten mit einer Auflösung von acht Bit; die Wandlungszeit beträgt typ. 15 us, im sample/hold-Betrieb ist eine Reduzierung auf 4 us möglich. Der D/A-Wandler ermöglicht die Ausgabe analoger Signale; hier beträgt die Wandlungszeit ca. 1 us.
Nach Angaben der Fa. SIKOS wird demnächst eine weitere A/D-Wandlerkarte für den ECB-Bus angeboten, die 12 Bit breite Informationen wandelt und in fast allen Funktionen softwareseitig steuerbar ist. Diese Karte ist ein Novum; eine separate Besprechung wird in einer der nächsten Ausgaben erfolgen.

Die Preise: (ohne MwSt)
- Interfaceplatine leer mit Manual ca. 69,- DM
- dto. mit Bauteilesatz ca. 172,- DM
- dto. Fertiggerät + 0,5 m Kabel ca. 294,- DM
- dto. Industrieversion ca. 383,- DM
- Busplatine mit 10 ECB-Slots, A-B-C-bestückt, 68000-fähig Leerplatine ca. 43,- DM
- Fertiggerät ca. 128,- DM
- Relaiskarte mit Manual, anschlußfertig ca. 299,- DM
- Optokopplerkarte mit Manual, anschlußfertig ca. 378,- DM
- Wandlerkarte mit Manual, anschlußfertig, Minimalstufe ca. 210,- / Maximalstufe ca. 430,- DM

Herstellung und Vertrieb:
Fa. SIKOS data-application, 8504 Stein

(ME)

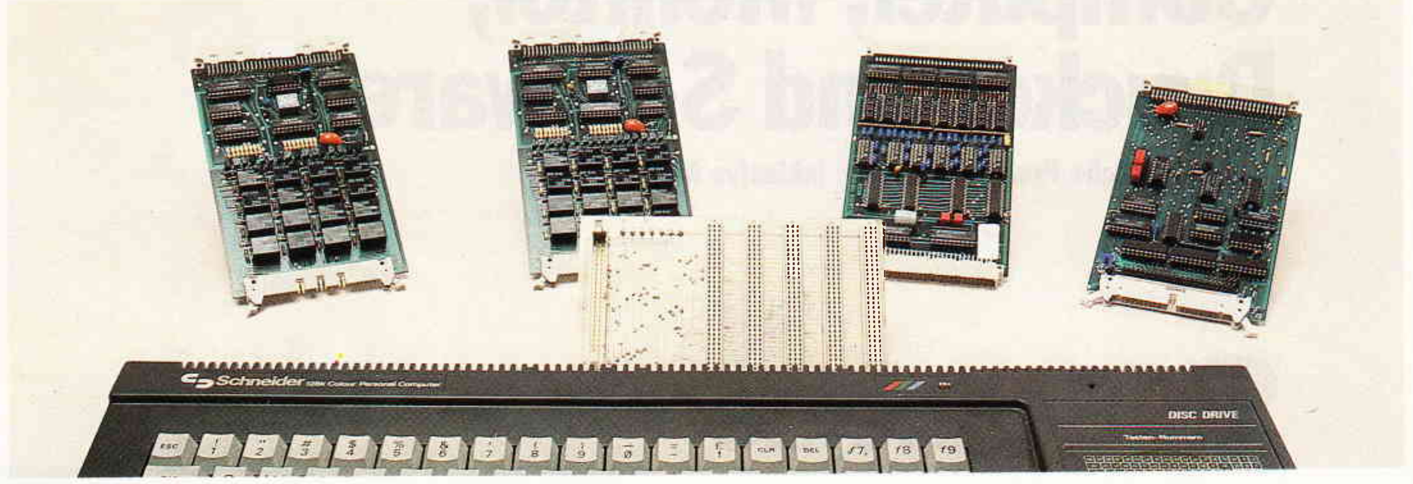


Bild 2: Die Karten am CPC



JOYCE

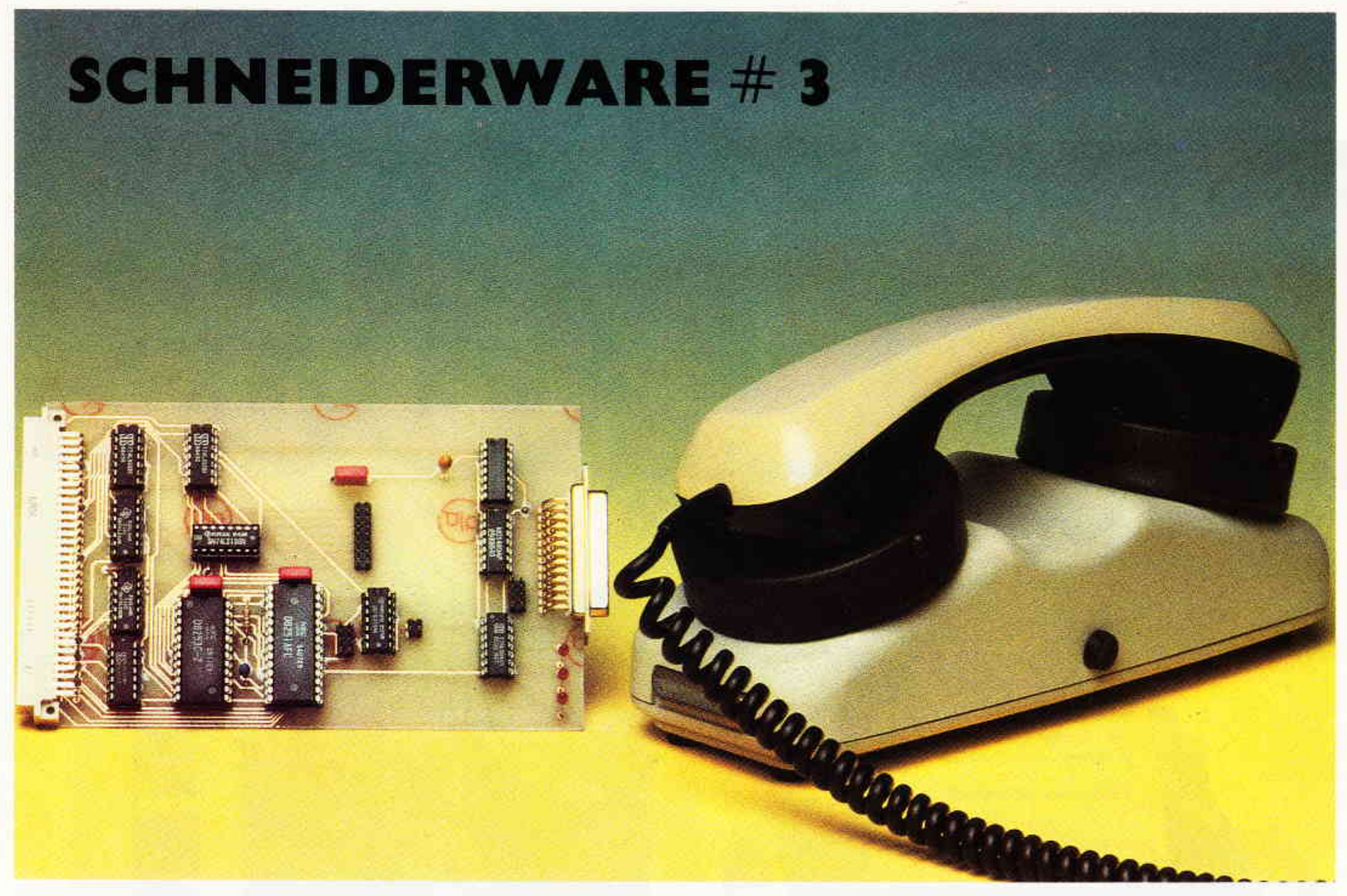
schon ab DM 1.799,–*

Das komplette Schreibsystem: Computer, Monitor, Drucker und Software.

* unverbindliche Preisempfehlung, inklusive MwSt.




ende...
Der Matrixdrucker von Schneider in Korrespondenzqualität
90 Zeichen/sec.-schnell
Schneider PCW 8256 JOYCE
Technische Informationen für alle, die vergleichen wollen
Arbeitsspeicher: 256 K RAM
Massenspeicher: 3" Compact Diskette 250 K (unformatiert)
Bildschirm: 32 Zeilen 90 Zeichen / Zeile
Tastatur: Deutsche Tastatur nach DIN (QWERTZ)
Software inkl.: Textverarbeitung "Locoscript", CP/M 3.0, Dr.LOGO, BASIC, GSX
JOYCE
PERSONAL COMPUTER TEXTSYSTEM
PCW 8256
Der Joyce von Schneider.
Und ab sofort sieht alles ganz anders aus:
Ein neues Schreibsystem, das mühelos in kürzester Zeit
Spitzen-Ergebnisse liefert.
Ein neuer Personal-Computer, der sich wirklich von Jedermann
ganz einfach bedienen läßt. Ein neues Text-Programm, das endlich
auf die Anforderungen zugeschnitten ist, die man in deutschen Büros
am häufigsten stellt.
Die Arbeit ist auf einmal viel effizienter. Und macht auch noch viel mehr Spaß.
JOYCE bringt die Wende.
Obwohl sein Preis eher eine Revolution ist ...
Komplett bedeutet für Sie:
Auspacken, aufbauen, anfangen.
Coupon:
An Schneider Computer Division, Silvastrasse 1, 8939 Türkheim
Schicken Sie die komplette JOYCE-Information kostenlos und unverbindlich an die richtige Adresse:
Name/Beruf:
Firma/Branche:
Strasse:
Ort:
CPC/8



SCHNEIDERWARE # 3

Hoffentlich ist Ihr Lötkolben vom Löten der Centronicschnittstelle noch heiß, denn heute geht es sofort weiter mit der Entwicklung einer universellen V/24-Schnittstelle. Ganz eiligen Lesern möchte ich kurz die technischen Daten bekanntgeben, bevor ich die unvermeidliche Theorie einer seriellen Datenübertragung erarbeite:
Baudrate (Übertragungsgeschwindigkeit) von 75 - 19600 Baud per Software einstellbar, getrennt für Sender und Empfänger, die Übertragungsparameter per RSX-Befehl wählbar, unabhängiges Ansprechen von 2 Karten durch entsprechende Adressdecodierung, Europakartenformat mit ECB-Anschlußbelegung sowie ein passendes Softwarepaket.

Einzeln ein- und aussteigen

Wir wollen uns noch einmal kurz ins Gedächtnis rufen, wie die parallele Datenverarbeitung funktionierte. Die Information liegt am 8 BIT Datenport zur gleichen Zeit an. Durch ein sogenanntes Übertragungssignal (STROBE) übernimmt dasjenige Bauteil die Daten, für das sie bestimmt waren. Das bedeutet, es werden acht verschiedene Informationen (BITS) gleichzeitig übertragen.

Die serielle Verarbeitung sieht nun etwas anders aus. Die Daten werden Bit für Bit hintereinander über ein, einem Schieberegister ähnlichen, Alleskönner ausgegeben. Das Prinzip können Sie aus Bild 1 ersehen. Ein großer Vorteil der seriellen Kommunikation ist der, daß zur Realisierung nur ein Leitungspaar erforderlich ist.

Denn die Bits werden ja nicht auf 8 Leitungen zur gleichen Zeit, sondern auf einer Leitung nacheinander übertragen. Dabei spielt die Übertragungsgeschwindigkeit, man nennt sie BAUDRATE (BITS / sec), eine entscheidende Rolle. Die Datenbits werden nun mit einer bestimmten Geschwindigkeit über unser Schieberegister ausgegeben (GESENDET).

Die Leitungen enden dann natürlich bei einem Empfänger, der unsere gesendeten Daten auch wieder empfangen sollte. Da er aber aus dem unendlichen Strom ankommender Daten nicht erkennen kann, wann ein Wort anfängt oder endet, haben sich pfiffige Leute eine sehr gute Lösung dieses Problems ausgedacht. Spätestens jetzt wird Ihnen klar, daß man zur seriellen Übertragung ein besonderes "ÜBERTRAGUNGSFORMAT" verwenden muß. Der Empfänger soll ganz klar erkennen können, wann ein Datenwort beginnt und wann es endet. Eine solche Aufgabe kann von einem simplen Schieberegister nicht mehr zufriedenstellend erfüllt werden.

Hochtrabende Namen

Die für den seriellen Datenverkehr angebotenen Bausteine tragen meist sehr exotische Namen. Hier gibt es 'USARTS', 'ACIAS' und 'UARTS '. Diese anspruchsvollen Namen erfüllen gerade den Anfänger mit ungeheuerer Ehrfurcht und tragen meist auch dazu bei, daß er lieber seinen Platz als Hardwareprofi anderen überläßt. Dabei sind diese Namen in der Regel nur die Abkürzungen der englischen Bausteinbeschreibungen.

Da die Funktionen derartiger IC's sehr ähnlich sind, wollen wir uns nur dem meistverwendeten Baustein, dem 'USART' 8251 widmen. USART heißt nichts anderes als 'U'niversal 'S'ynchron 'A'synchron 'R'eciver (Empfänger) 'T'ransmitter; welch ein Satz. Ich habe diesen Baustein allerdings nicht wegen seines schönen Namens ausgewählt, sondern wegen seiner Leistungsfähigkeit und der einfachen Bedienung, die Ihnen später sicher zugute kommen wird.

BIT für BIT

Vom Prinzip ist so ein 'USART' auf

der Senderseite ein Parallel/Seriellwandler und auf der Empfängerseite ein Seriell/Parallelwandler, d.h. er zerstückelt die vom Microcomputer gelieferten Daten (8 BIT WORT) und gibt sie kleckerweise aus. Beim Entgegennehmen der seriell ankommenden Daten setzt er sie umgekehrt zusammen und übergibt sie wieder als 8 Bit Wort dem Prozessor. Nun ist es sehr nützlich zu wissen, wann ein Datenwort zu Ende ist und wann ein neues anfängt; andernfalls gibt es unweigerlich Bitsalat. Für das 'PROTOKOLL' sind bestimmte Vereinbarungen getroffen worden. Die Sende- und Empfangsleitungen befinden sich in Ruhe immer in einer Markierungsstellung (MARK), das bedeutet, sie führen HIGH-Pegel. Nun quetscht man (bzw. der 8251 tut das von sich aus) vor das Datenwort ein zusätzliches Bit, welches immer LOW-Pegel führt. Dieses BIT nennt man 'STARTBIT'. An der abfallenden Flanke dieses Startbits erkennt der Baustein den Beginn des Datenwortes und beginnt, die nachfolgenden Bits zu zählen.

Nach dem 7-ten oder 8-ten Bit (je nach Programmierung des 8251) fügt der 8251 noch ein sogenanntes Stopbit an. Der Pegel dieses Bits entspricht wieder dem Zustand MARK (HIGH). Der Baustein hat noch die Möglichkeit, die Anzahl der Stopbits festzulegen; 1, 1. 5 oder 2 sind möglich. Mit diesem Wissen können Sie schon beginnen, vorausgesetzt, die entsprechende Hardware ist vorhanden, um Daten auf die lange Reise zu schicken. Technisch funktioniert das einwandfrei. In Bild 2 sehen Sie die schematische Darstellung eines Datenwortes mit Start und Stopbits. Bei Übertragungen, die eine hohe Datensicherheit verlangen, besitzt der 8251 eine Einrichtung, die es dem Programmierer ermöglicht, eine Prüfung der ankommenden Daten vorzunehmen. Diese nützliche Austattung heißt 'PARITYPRÜFUNG'. Dieses Testverfahren verwendet ein internes Bit, genannt 'PARITYBIT'. Dieses Verfahren kann allerdings nur einen aufgetretenen Fehler erkennen, ihn aber nicht richtigstellen. Die Paritprüfung funktioniert also folgendermaßen:

Der 8251 zählt nun während der Übertragung die Anzahl der binären Einsen, die in einem Datenwort vorhanden sind. In Abhängigkeit von dieser ermittelten Zahl setzt der Usart ein Bit auf HIGH und fügt es nach dem Datenwort in den Bitstrom ein. Der Empfänger zählt nun wieder die

Bild 1: Schema serielle und parallele Übertragung

D7	D6	D5	D4	D3	D2	D1	D0
0	1	0	1	0	0	1	1

ZU ÜBERTRAGENDES DATENWORT

Einsen und erzeugt das Paritybit, vergleicht es mit dem Mitgeschickten und setzt ein entsprechendes Bit in einem Register (Speicherzelle), entsprechend dem Ergebnis. Ebenso kann der 8251 sogenannte 'FRAMING ERRORS' (Rahmenfehler, kein Stopbit vorhanden) oder sogenannte 'OVERRUN ERRORS' (Überlauffehler, entsteht dann, wenn ein Zeichen empfangen wurde, das frühere von der CPU noch nicht abgeholt wurde, also verlorenging), erkennen und sie dem Programm mittels dieses Registers mitteilen. Dieser Möglichkeiten werden wir uns nicht bedienen, da unsere Daten nicht von so wichtiger Natur sind, um derart überwacht zu werden.

Unser USART wird aktiv

Da der 8251 aufgrund grauer Theorie nicht zu arbeiten beginnen kann, müssen wir endlich aktiv werden.

Wie Sie im letzten Beitrag gelesen haben, übergibt man diesen hochkarätigen Schaltkreisen ihre Arbeitsvorgaben mittels Steuerwörtern in bestimmten Registern. Der 8251 hat deren 2. In diesen Bytes, es können durchaus mehrere sein, teilt man dem Baustein mit, was er in Zukunft zu tun hat. Nach dem Einschalten der Versorgungsspannung von 5 Volt, wie sie für TTL-Bausteine zwingend vorgeschrieben ist, befindet sich der 8251 in einem undefinierten 'Wartezustand'. Der Baustein 8251 besitzt zwei Register (Speicherzellen), ein Register für die Steuerwörter und eins zum Abspeichern der gesendeten und empfangenen Bytes. Nachdem das Steuerregister an der Adresse &F9E1 nach dem Einschalten einen zufälligen Wert annimmt, weiß der Baustein nicht, was er zu tun hat. Deswegen übergeben wir ihm erst einmal drei 'Nullbytes' in Form folgender Befehle: OUT &F9E1,0:OUT &F9E1,0: OUT &F9E1,0:und befehlen ihm, erst einmal 'nichts' zu tun. Auf jeden Fall hat er die Anzahl der Bytes, die wir ihm übergaben, mitgezählt. Jetzt wird ein Befehlsbyte erwartet, und dieses übergeben wir ihm wieder in Form eines Out &F9E1,&40. Dadurch wird der 8251 intern in einen definierten Zustand versetzt und wir können zu arbeiten beginnen. Nach einem 'INTERNEN RESET', wie dieser Befehl heißt, interpretiert der Chip das nächstfolgende Byte als 'MODUSWORT'. Da er aber nur 2 Register zur Verfügung hat und zum SENDEN und EMPFANGEN schon eines benötigt, bleibt uns nichts anderes übrig, als das Moduswort und das Befehlswort nacheinander zu übergeben. Der 8251 erwartet während der Initialisierungsphase zuerst das Moduswort (Leitung C/D auf HIGH). Das Byte, das unmittelbar auf dieses Moduswort folgt, wird als Befehlswort interpretiert. Mit dem Moduswort legen wir die Anzahl der Datenbits, der Stopbits, den Modus der Paritypüfung und den Baudratenfaktor, der die Werte 1,16 und 64 annehmen kann, fest. Dieser Faktor teilt die Frequenz, die an den Pins RxC und TxC durch eben diese drei Werte. Mit Hilfe des Befehlswortes, das innerhalb eines Programms auch mehrmals geändert werden kann, gibt man den Sender und den Empfänger frei oder sperrt ihn. Ebenso kann man bestimmte Pins des 8251 entsprechend setzen (HIGH) oder löschen (LOW), wie es der Programmablauf gerade erfordert. Auch wird der interne Reset (BIT 7 auf HIGH) eingeleitet. Bit 4 setzt die Meldungen der Fehler zurück (Parity, Frame und Overrun). Die Tabellen 1a bis c geben Ihnen eine Übersicht der Programmierungsmöglichkeiten. Nun ist es langsam an der Zeit, den Baustein mit unserem Microprozessor zu verbinden.

Der Z 80 steuert einen USART

Da Sie mittlerweile zu ausgefuchsten Hardwareprofis heranreifen, möchte ich Sie mit den alten Kamellen der Adressdecodierung nicht mehr langweilen, sondern gleich richtig einsteigen. Aus dem Schaltplan, den Sie in Skizze 2 bewundern können, sind Sie sicher in der Lage, die Adressdecodierung gemäß Schneiderware #1, nachzuvollziehen. Sie finden die entsprechenden Portadressen in Tabelle 2. Viel interessanter für uns sind jetzt die Anschlüsse des 8251, die ihn zum Leben erwecken. Bild 3 macht Sie mit deren Anschlußbelegung vertraut. Anschlüsse, deren Namen wir schon

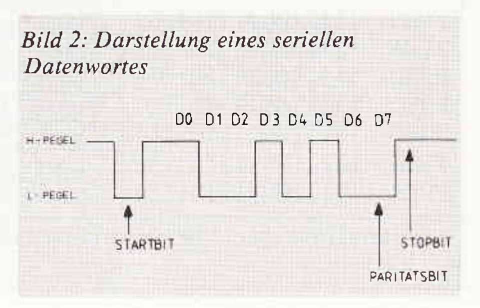


Bild 2: Darstellung eines seriellen Datenwortes

Tab. 1: Moduswort 8251

D7 D6 D5 D4 D3 D2 D1 D0

D2 – D0 NEHMEN DEN GLEICHEN LOGISCHEN ZUSTAND AN, WIE DIE ENTSPRECHENDEN ANSCHLÜSSE DES 8251:
D2 = TxE ; D1 = RxRDY ; D0 = TxRDY

PARITÄTSFEHLER
L = OHNE FEHLER
H = ES WURDE EIN FEHLER BEI EINEM EMPFANGENEN ZEICHEN ERKANNT
ES KANN MIT DEM BIT D 4 DES KOMMANDO-STEUERWORTES ZURÜCKGESETZT WERDEN.

ÜBERLAUFFEHLER
L = OHNE FEHLER
H = DER INHALT DES PUFFERS WURDE ÜBERSCHRIEBEN, BEVOR DER RECHNER DIE ZEICHEN AUSLESEN KONNTE.

RAHMENFEHLER
L = OHNE FEHLER
H = AM ENDE EINES DATENWORTES KONNTE KEIN STOPPBIT ERKANNT WERDEN. DIESER FEHLER IST NUR IM ASYNCHRONEN BETRIEB MÖGLICH. BIT D4 SETZT DIESE FEHLERKENNUNG ZURÜCK

DIE BITS D7 UND D6 GEBEN DEN JEWEILIGEN PEGEL DES 8251 WIEDER

Befehlswort 8251

ASYNCHRON

D7 D6 D5 D4 D3 D2 D1 D0

BAUDRATENFAKTOR

D1	D0	BEDEUTUNG
L	L	SYNCHRON - BETRIEB
L	H	FAKTOR 1
H	L	FAKTOR 16
H	H	FAKTOR 64

ZEICHENLÄNGE (DATEN)

D3	D2	BEDEUTUNG
L	L	5 BIT ZEICHENLÄNGE
L	H	6 BIT ZEICHENLÄNGE
H	L	7 BIT ZEICHENLÄNGE
H	H	8 BIT ZEICHENLÄNGE

PARITÄTSFREIGABE
H = FREIGEBEN (EIN) L = SPERREN (AUS)

PARITÄTSPRÜFUNG
H = GERADE PARITÄTSPRÜFUNG L = UNGERADE PRÜFUNG

ANZAHL DER STOPP - BITS

D7	D6	BEDEUTUNG
L	L	UNGÜLTIG !
L	H	1 BIT
H	L	1,5 BIT
H	H	2 BIT

Kontrollwort 8251

D7 D6 D5 D4 D3 D2 D1 D0

SENDEFREIGABE
L = SENDER AUS
H = SENDER EIN

SETZEN DATA TERMINAL READY (DTR)
H = DTR - AUSGANG WIRD AUF NULL GESETZT

SETZEN EMPFANGSFREIGABE
L = DATENEMPFANG IST GESPERRT
H = DATENEMPFANG IST GEGEBEN

"BREAK" ZEICHEN SENDEN
L = NORMALER BETRIEB
H = AUSGANG TXD WIRD AUF L-PEGEL GESETZT, NORMALFALL = H-PEGEL

FEHLERKENNUNG
H = ALLE FEHLERKENNUNGEN WERDEN ZURÜCKGESETZT: PARITÄTS-, ÜBERLAUF-, ZEICHENFEHLER IM STEUERWORT ZUSTANDSEINLESUNG

SENDEAUFFORDERUNG
H = DER RTS-AUSGANG WIRD AUF H GESETZT

INTERNES RÜCKSETZEN
H = NÄCHSTES KOMMANDO IST MODUSWORT, BEDINGUNG C/D = 1

BETRIEBSART " SUCHEN "
H = NUR BEI SYNCHRONER ARBEITSWEISE, ERMÖGLICHT SUCHE NACH DEM SYNCHRONISATIONSZEICHEN

Tab. 1: Steuerwort 8253

ZÄHLERAUSWAHL		LESEN / LADEN		BETRIEBSART			ZÄHLART
D 7	D 6	D 5	D 4	D 3	D 2	D 1	D 0

ZÄHLART
1 = DEZIMAL, MAX 10^4
0 = BINÄR , MAX 2^{16}

BETRIEBSART (MODE)

D3	D2	D1	BEDEUTUNG:
L	L	L	MODE 0: INTERRUPT BEIM LETZTEN TAKT
L	L	H	MODE 1: PROGRAMMIERBARER ZEITGEBER
L	H	L	MODE 2: TAKTGENERATOR
L	H	H	MODE 3: RECHTECKGENERATOR
H	L	L	MODE 4: GETRIGGERTER IMPULS-SOFTWARE
H	L	H	MODE 5: GETRIGGERTER IMPULS-HARDWARE

LESE / LADEMODUS

D5	D4	BEDEUTUNG:
L	L	DER AUGENBLICKLICHE ZÄHLERSTAND WIRD ABGESPEICHERT UND KANN VON DER CPU GELESEN WERDEN
L	H	LESEN UND SCHREIBEN DES NIEDERWERTIGSTEN BYTES
H	L	LESEN UND SCHREIBEN DES HÖHERWERTIGEN BYTES
H	H	LESEN UND SCHREIBEN BEIDER BYTES, DAS NIEDERWERTIGE ZUERST

ZÄHLERAUSWAHL

D7	D6	BEDEUTUNG:
L	L	ZÄHLER-0
L	H	ZÄHLER-1
H	L	ZÄHLER-2
H	H	UNGÜLTIG

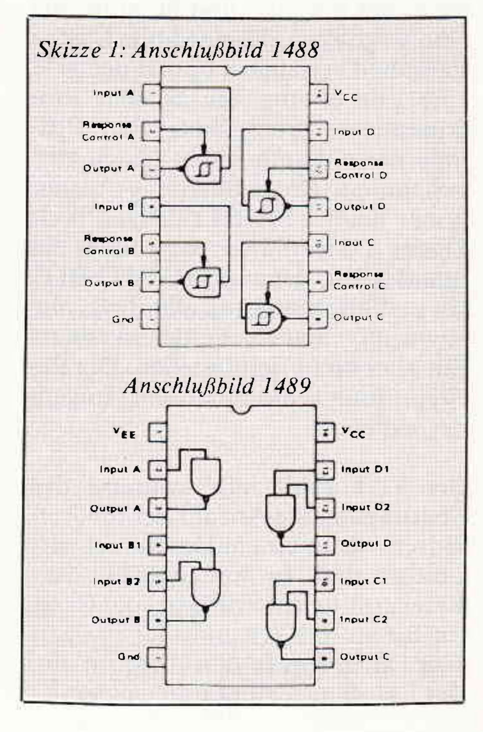
kennen (D0-D7,RD,WR,und CS), haben bei diesem Baustein die gleiche Funktion wie z.B beim 8255 aus dem letzten Beitrag. Da ich nicht alle Beinchen dieses vielseitigen 'Käfers' besprechen kann (würde allein ein ganzes Heft füllen), möchte ich mich auf die Pins beschränken, die für uns wichtig sind. Da sind die Anschlüsse 'RxD' und 'TxD'.

Über diese beiden Leitungen werden die Daten gesendet (TxD Transmit Data) und empfangen (RxD Receive Data). Eine weitere wichtige Leitung ist der Pin 17 des 8251 (CTS Clear to Send - Klar zum Senden). Die deutsche Übersetzung sagt eigentlich schon alles über deren Bedeutung aus. Sie wird vom Baustein ähnlich verwendet, wie die Leitung BUSY, die uns von der Centronics schon bekannt ist. Wenn dieses Signal auf LOW ist, dann kann der Baustein senden, vorausgesetzt, der Sender wurde mittels Befehlsbyte (bit 0 auf 1) enabled (freigeschaltet). Auch ein serieller Drucker verwendet dieses Signal, um den sogenannten Handshake (Händeschütteln)-betrieb zu ermöglichen. Ferner gibt es noch eine Reihe anderer Pins, deren Zustände über das Controllwort (print(inp(&f9e1))) abgefragt werden können. Für unsere Software sind allerdings nur die Bits TxRDY (Sendebuffer leer=1) und RxRDY (Empfangsbuffer voll=1) von Bedeutung. Diese Abfrage benötigen wir später in unserem Ansteuerprogramm. Nun bleiben noch drei Pins (clock-20,RxC-25 und TxC-9). Pin 20 Clock wird einfach mit dem Prozessortakt verbunden (4MHZ). Die beiden anderen möchte ich erst etwas später besprechen.

Codiermöglichkeiten der V/24 - Karte — *Tab. 2: Portadressen der V/24*

Adresse	Binär	Funktion	Karte	Einstellung	Jumper
F9E0	1111100111100000	Daten	Karte 1	vorein-	J7-7
F9E1	1111100111100001	Steuerwort	Karte 1	gestellt	Q0
F9E2	1111100111100010	Daten	Karte 2	Brücke	J8-8
F9E3	1111100111100011	Steuerwort	Karte 2	trennen	Q1
F9E4	1111100111100100				J9-9
F9E5	1111100111100101		frei		Q2
F9E6	1111100111100110				J10-10
F9E7	1111100111100111		frei		Q3
F9E8	1111100111101000	Timer 0	Karte 2	Q4+Q5	J11-11
F9E9	1111100111101001	Timer 1	Karte 2	stecken	Q4
F9EA	1111100111101010	Timer 2	Karte 2	Q6+Q7	J12-12
F9EB	1111100111101011	Steurwort	Karte 2	trennen	Q5
F9EC	111110011110110	Timer 0	Karte 1	vorein-	J13-13
F9ED	1111100111101101	Timer 1	Karte 1	gestellt	Q6
F9EE	1111100111101110	Timer 2	Karte 1		J14-14
F9EF	1111100111101111	Steuerwort	Karte 1		Q7

Einstellung der Vorteilerfrequenz: 2 MHz voreingestellt J1-1
1 MHz J2-2



Skizze 1: Anschlußbild 1488

Anschlußbild 1489

Mit der Zeit im Clinch

In sehr vielen Anwendungsfällen ist ein Microcomputer nur damit beschäftigt, seine Zeit damit zu verschwenden, auf das eine oder andere Ereignis eine bestimmte Zeit zu warten. Dabei gibt es eine Reihe unterstützender Bausteine, die unserem Microprozessor solche Nebentätigkeiten abnehmen können. Diese Zeitgeber oder Timer sind geradezu prädestiniert, der CPU bei solchen Arbeiten kräftig unter die Arme zu greifen. Ein typischer Vertreter aus der Familie der Timer ist der 8253 von INTEL. Dieser Ic besitzt drei unabhängig voneinander arbeitende Rückwärtszähler, deren ausgangsseitiges Verhalten sehr vielseitig programmiert werden kann. Jeder Zähler (Timer) besitzt einen eigenen Takteingang (CLOCK), der mit einem dazugehörigen Freigabeingang (GATE) UND verknüpft ist, d.h. Taktimpulse bewirken nur dann ein Zählen, wenn der GATEeingang auf HIGH-Pegel ist. Die Zählmethode (BINÄR oder BCD), sowie das Verhalten der Ausgänge kann für jeden Zähler separat mittels Steuerwort programmiert werden. Um nun den Timer zum Arbeiten anzuregen, übergeben wir ihm an den Adressen &F9EF ein Steuerwort und laden nun an den Datenadressen der jeweiligen Zähler (&F9EC-&F9EE) die Werte, bei denen sie beginnen sollen zu zählen. Sie werden sich fragen, was das eigentlich soll: Weiter oben machte ich Sie mit den Signalen RxC und TxC bekannt; an dieser Stelle beleuchten wir sie näher. Der Eingang RxC ist die Frequenz, die die Empfangsgeschwindigkeit einstellt, analog dazu legt der Eingang TxC die Geschwindigkeit beim Senden des Bausteins fest. Wir möchten nun mittels Timer diese Baudraten (Frequenzen) beliebig verändern, und das natürlich per Software.

Tab. 3: Baudratentabelle

BEZUGSFREQUENZ & TIMERWERT 1 MHZ	2 MHZ	FREQUENZ KHZ	TEILUNGSFAKTOR 8251: 1	16	64
(6)	(13)	153,6 KHZ	-	9200	2400
(13)	(26)	76,3 KHZ	-	4800	1200
(26)	(52)	38,4 KHZ	-	2400	600
(52)	(104)	19,2 KHZ	19200	1200	300
(104)	(208)	9,6 KHZ	9600	600	150
(208)	(416)	4,8 KHZ	4800	300	75
(416)	(833)	2,4 KHZ	2400	150	37,5
(833)	(1666)	1,2 KHZ	1200	75	-
(1666)	(3333)	0.6 KHZ	600	37,5	-

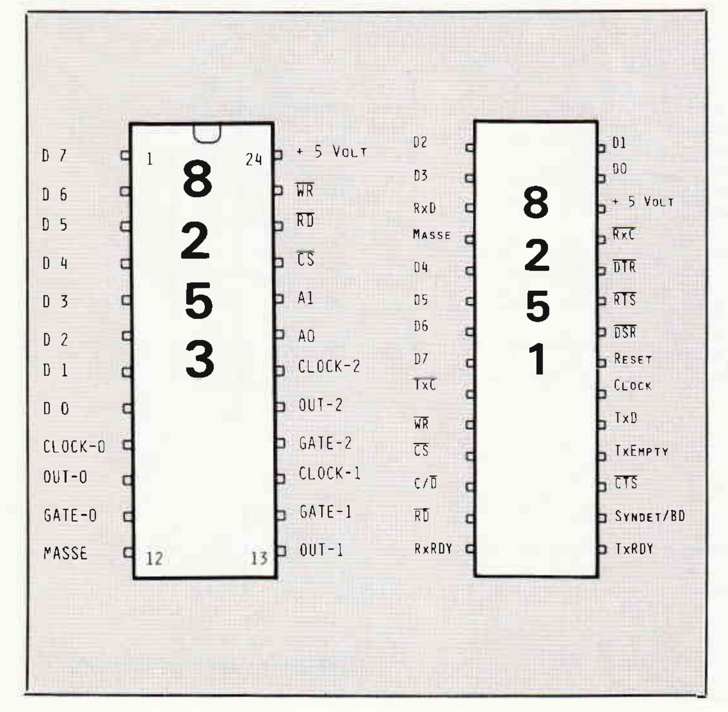
ZWISCHENWERTE KOENNEN JEDERZEIT DURCH VERAENDERN DER ZEITKONSTANTEN ERZEUGT WERDEN.

Wohldosierte Bits

Der Einsatz eines Zeitgeberbausteins reicht vom programmierbaren Rechteckgenerator bis hin zum Langzeittimer, der Signale erzeugen kann, die sich nur alle paar Jahre wiederholen. So lange wollen wir aber nicht auf unseren seriellen Datenstrom warten.

Da unsere Bits einzeln und nacheinander auf die Reise gehen, benötigen wir einen Taktgeber, der die Bits schrittweise an die Ausgänge durchschiebt. Die Geschwindigkeit, mit der das Ganze geschieht, nennt man wie sie schon wissen 'BAUDRATE'; das ist die Anzahl der Bits, die in einer Sekunde übertragen werden. Bei unserer Baudrate von 300 Baud werden sich in einer Sekunde 300 Bits durch den Draht quälen müssen. Das ist schon eine beachtliche Leistung. Als weit verbreitete Baudrate sind 9600 Baud in großen Systemen eine durchaus übliche Geschwindigkeit. Unsere Schnittstelle kann dank Timern ebenfalls höhere Geschwindigkeiten erreichen (z.B. Ansteuern eines seriellen Druckers mit 1200 baud) ist durchaus möglich. Eine Besonderheit weist diese V/24-Schnittstelle auf: Durch die Möglichkeit, die Geschwindigkeit für Senden und Empfangen getrennt einzustellen, steht Ihnen mit einem geeigneten Akustikkoppler die Welt des BTX offen. Diese Übertragungsart sendet mit 1200 Baud und empfängt auf dem Rückkanal mit einer Baudrate von 75 Baud – für uns kein Problem.
Um die entsprechenden Startwerte in den Timer zu bekommen, ist etwas Mathematik nötig. Wie oben schon erwähnt, zählt der Timer rückwärts, d.h. er subtrahiert ausgehend von einem Anfangswert immer 1 bei je-

Bild 3: Anschlußbild 8251/8253




```
5 MODE 0:PRINT CHR$(24)"V / 24 T E S T E R [3099]
 "CHR$(24)
6 PRINT:PRINT:PRINT:PRINT:PRINT "(C)  RICH [8916]
TER PETER ":PRINT:PRINT:PRINT:PRINT:PRINT
CHR$(24)"PARAMETER SIND FEST EINGESTELLT"C
HR$(24)
7 PRINT:PRINT:PRINT CHR$(24)"300 BAUD,8 BI [5720]
TS,NO  PARITY " CHR$(24):LOCATE 1,25:PRINT
 "TASTE DRUECKEN ":CALL &BB18:MODE 2
10 CLS:INPUT "DRUCKER EIN (1) ODER AUS (0) [3442]
   ",DRUCKER
11 IF DRUCKER =1 THEN DRUCKER =-1 ELSE DRU [1405]
CKER =0
50 MODE 2                                  [513]
100 IF PEEK(&9000)=0 THEN MEMORY &5FFF:LOA [6091]
D"mail.bin",&9000:GOTO 200
200 CALL &9000:|BAUD,416,416:|SIOINIT,8,0, [2070]
2,0
500 DEFINT a                               [357]
510 CLS                                    [91]
1000 ein=-1:aus=0                          [653]
1010 a$=INKEY$:IF a$<>"" THEN GOSUB 1050   [2595]
1020 a=0:|RECV,@a:IF a<>0 THEN GOSUB 1090  [3584]
1030 GOTO 1010                             [345]
1050 IF a$=CHR$(12) THEN CLS:RETURN        [1827]
1060 |SEND,ASC(a$)                         [1315]
1080 RETURN                                [555]
1090 PRINT CHR$(a);                        [1200]
1100 IF drucker=EIN THEN |PRINT,a          [945]
1110 RETURN                                [555]
```

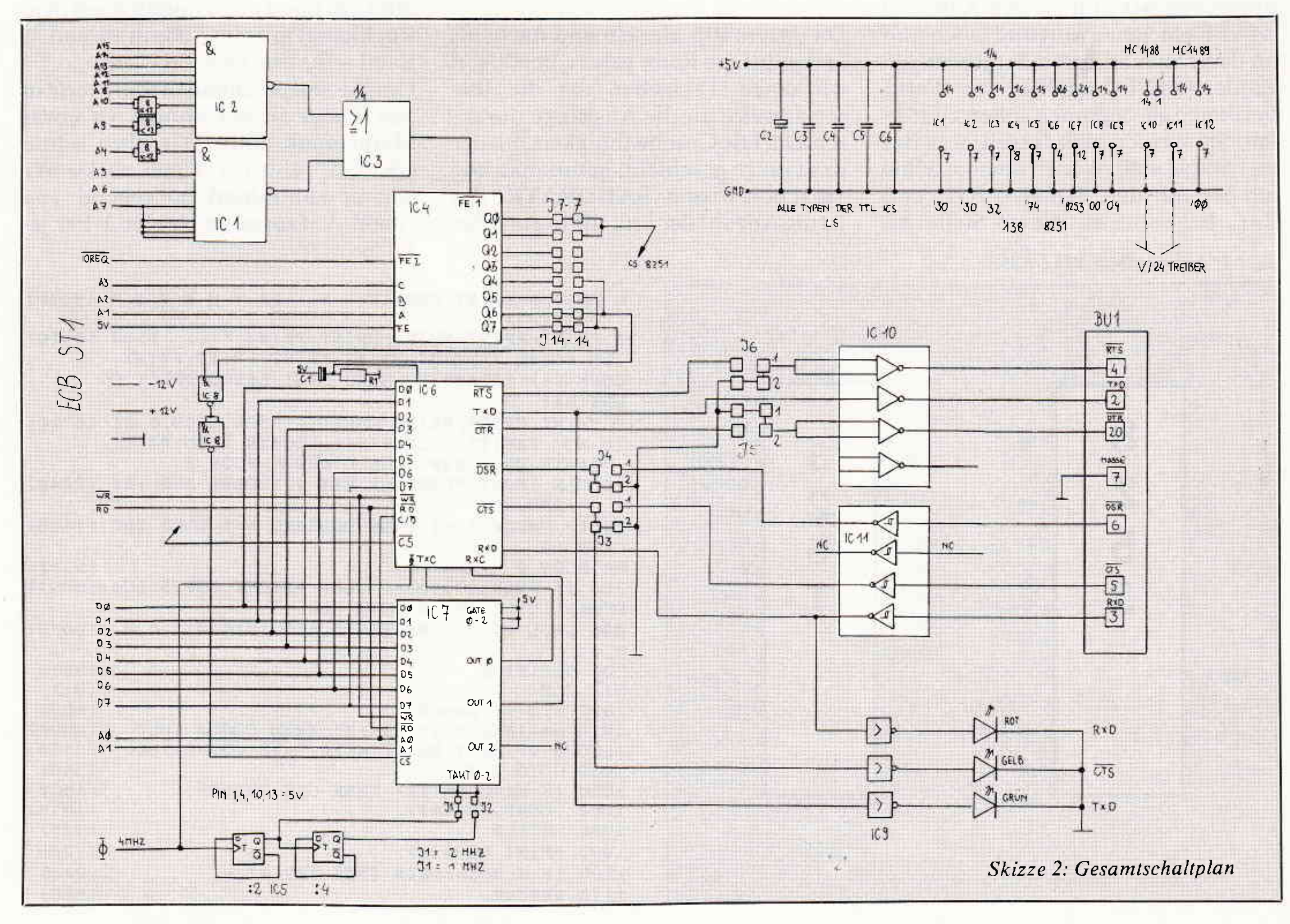
dem Taktimpuls des Clockeingangs. In dem Augenblick, da der Zähler 0 beinhaltet, gibt er einen Impuls am Ausgang ab, der die Länge einer Taktperiode des Clocksignals besitzt. Skizze 4 zeigt Ihnen diesen Zusammenhang, der für den Modus 2 zutrifft. Da dieses Signal einen unsymmetrischen Charakter besitzt (High Zustand dauert länger als LOW), benutzen wir für unsere Anwendungen den MODE 3. Hier ist das Ausgangssignal genau symmetrisch (HIGH und LOW zu gleichen Teilen). Den Zählerwert, den Sie nun programmieren müssen, um eine gewünschte Ausgangsfrequenz zu bekommen, errechnen sie nach folgender Formel: Periodendauer = 1/ Frequenz.

Ein Beispiel: Für eine Baudrate von 300 Baud benötigen Sie eine Frequenz (Tabelle 3) bei einem Teilungsfaktor (des 8251) vnn 16 entspricht dies einer Frequenz von 4.8 KHz. Eine kurze Rechnung mit dem Taschenrechner bestätigt uns den Wert von t = 1 / 4800 = 208 us. Diesen Wert müssen Sie dem Befehl Ibaud, 208, 208 zweimal übergeben, einmal für die Empfangsrichtung, und einmal für die Senderichtung. Dabei ist die Brücke J2 gesteckt, das bedeutet eine Eingangsfrequenz von 1MHz. In der Stellung J1 besitzt die Eingangsfrequenz den doppelten Wert (immer nur eine Brücke stecken!). Mit dieser Frequenz hat der Zähler im Timer schon in der Hälfte der Zeit den Nulldurchgang erreicht und gibt den gewünschten Impuls ab. Um in diesem Fall das gewünschte Ergebnis zu erhalten, nehmen wir den Wert 208 einfach mal 2 und veranlassen den Zähler, doppelt lange zu zählen. Der Eingangstakt wird einfach aus dem Prozessortakt am Pin 29c des Kartensteckers entnommen und dann mit einem Teiler durch 2 geteilt. Dieser Teiler besteht aus zwei hintereinandergeschalteten D-Flipflops vom Typ 74LS74. Dieser IC erspart uns einen teuren Quarzgenerator.

Die Hardware ist des Ganzen Anfang

Nachdem wir uns nun ausführlich mit den wichtigsten Bausteinen unserer V/24-Schnittstelle beschäftigt haben, schreiten wir zum Aufbau der Platine. Entweder Sie haben sich die Platine selbst hergestellt oder von der Möglichkeit unseres Platinenservice Gebrauch gemacht. In beiden Fällen sollten Sie zuerst die nötigen Brücken einlöten. Die Ic's gehören in gute Fassungen. Den Widerstand und die Kondensatoren (auf richtige Polung bei den Elektrolyttypen achten) kommen als nächstes dran.

Nach dem Einlöten der Bauteile stecken Sie die Karte OHNE ICS erstmalig an Ihren CPC. Sollten nach dem Einschalten kleine Rauchwölkchen aufsteigen, würde ich eine nochmalige Kontrolle Ihrer Arbeit empfehlen (kleine Perlen, die evtl. Leiterbahnen kurzschließen). Wenn allerdings das gewohnte 'READY' auf dem Monitor Ihres CPC's erstrahlt, können Sie getrost die restlichen Bauteile in ihre Fassungen stecken, bei ausgeschalteter Spannung versteht sich, und die gleiche Prüfung wiederholen. Die zahlreichen Einstellmöglichkeiten an Ihrer Karte wurden schon bei der Erstellung des Layouts berücksichtigt und auf Standardwerte voreingestellt. Darum brauchen Sie sich nicht mehr zu kümmern. Wenn Sie Ihre Karte allerdings universell verwenden möchten, empfehle ich Ihnen sogenannte Steckbrücken einzulöten. In diesem Fall müssen Sie auf der Unterseite der Platine die voreingestellten Brücken auftrennen und sie durch Ihre Einstellung ersetzen. Das dürfte aber mit Hilfe des Schaltplanes keine Schwierigkeiten bereiten.



Skizze 2: Gesamtschaltplan



Platinenservice

Die SCHNEIDERWARE ist ein universelles Erweiterungssystem für alle CPC's auf der Basis des bekannten ECB-Bussystems. Dieses System läßt sich mit minimalen Adaptionsmaßnahmen an alle Rechner mit Z80-CPU anschließen. Für Ihren CPC brauchen Sie zunächst:

1. Das passende Kabel (im Artikel beschrieben; wird im Platinenservice angeboten)
2. Die Basisplatine (zur Umsetzung der Pinbelegung CPC/ECB).

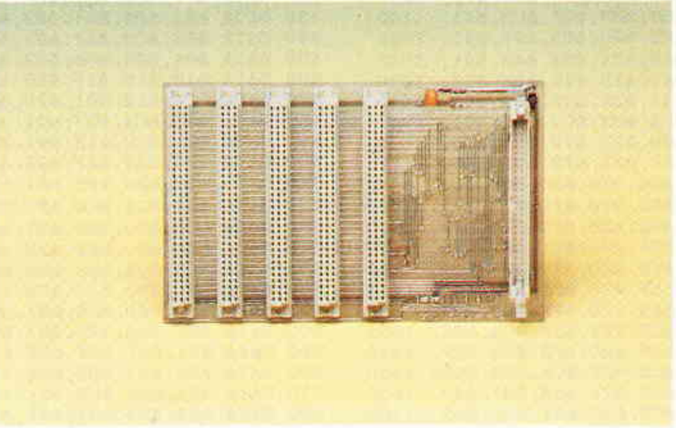
Im Lauf der Serie wird für Einsteiger der Aufbau eines 19"-Systems und die Integration der Karten in dieses Gehäuse beschrieben. Die Basisplatine besteht aus beidseitig kupferbeschichtetem glasfaserverstärktem Epoxydharz und ist durchkontaktiert. Die Einsteckkarten sind einseitig beschichtet und aus dem gleichen Material wie die Basisplatine. Die in den Fertiggeräten verwendeten Bausteine sind von bester Qualität; die fertigen Geräte sind geprüft. Die in Heft 7 vorgestellte Centronics-Parallelschnittstelle ist gegenüber dem CPC-Druckerport vollständig und vollkompatibel zu allen Geräten mit Centronics-Anschluß.

Zahlungsbedingungen:

Gesamtpreis zuzüglich 5,– DM Porto/Verpackung (im Ausland 8,– DM Porto/Verpackung)

Am einfachsten per Vorkasse (Verrechnungsscheck) oder als Nachnahme zuzügl. der Nachnahmegebühr (in das Ausland nicht möglich).

Bitte Postkarte im Heft benutzen!

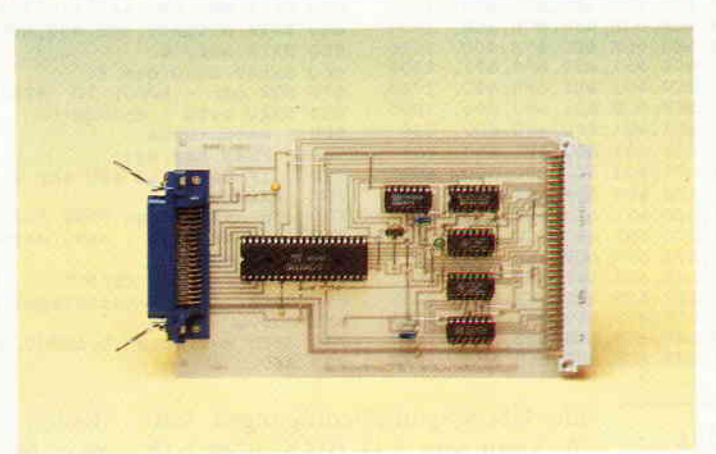


Kabel und Basisplatine

Ein 50-poliges Kabel mit den passenden Steckern für Ihren CPC und die Basisplatine mit Steckplätzen für max. 5 Erweiterungskarten stellen den Grundstock für Ihre Hardware-Erweiterungen dar. Die Basisplatine adaptiert den Erweiterungsport auf die Anschlußbelegung des weitverbreiteten ECB-Bussystems.

Die Preise:

Kabel für 464/664	**35,90**
Kabel für 6128	**45,90**
Platine, unbestückt	**24,90**
Karte, geprüft	**62,90**



Centronics

Diese Karte ist eine vollwertige 8-Bit-Parallelschnittstelle nach Centronics-Standard. Jegliche kommerzielle Peripherie arbeitet problemlos mit dieser Karte zusammen. Die Software zu dieser Schnittstelle ist in Heft 7/86 abgedruckt. Die Karte kann direkt in ein ECB-Bussystem eingesteckt werden und ist mit angepaßter Software auf allen Z80-Rechnern lauffähig.

Die Preise:

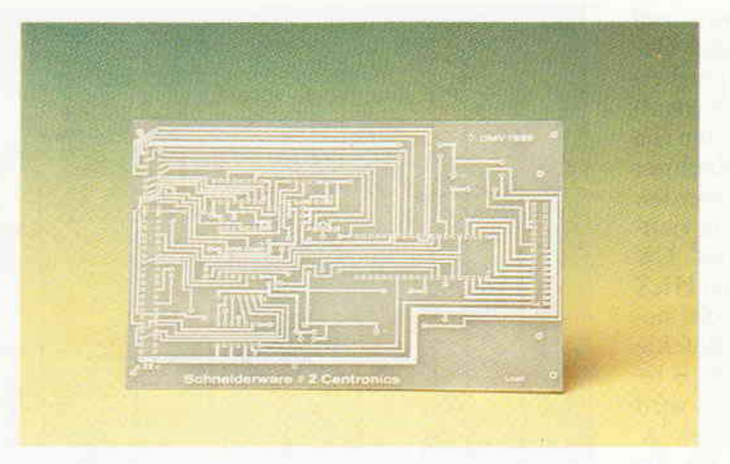
Platine, unbestückt	**17,90**
Karte, geprüft	**79,90**

V/24

SCHNEIDERWARE #3 präsentiert eine komplette V/24 (RS 232)-Schnittstelle. Auch diese Karte ist, wie sämtliche Peripherie aus dieser Serie, voll kompatibel zu kommerziellen Geräten mit RS 232-Schnittstelle. Die Steuerbefehle sind als RSX-Befehle gestaltet und können problemlos in eigene Basicprogramme eingebunden werden. Die Bauanleitung und die Software dieser universellen Schnittstelle ist in CPC Schneider International, Heft 8/86 abgedruckt.

Die Preise:

Platine, unbestückt	**29,80 DM**
Karte, geprüft	**139,90 DM**



Platine, unbestückt

SCHNEIDERWARE ist in drei Versionen für Sie verfügbar. Sie können nach Bauplan selbst bauen, die fertig bestückten und geprüften Karten über den Platinenservice erhalten oder die unbestückte Platine erwerben. Diese werden in Industriequalität gefertigt, sind verzinnt und gebohrt; doppelseitig beschichtete Platinen sind chemisch durchkontaktiert und geprüft. Hierbei haben Sie den Vorteil, die Platine nicht selbst herstellen zu müssen, jedoch die Bestückungskosten zu sparen und die Bauteile selbst einzukaufen.

Schneider CPC International

Postfach 250, 3440 Eschwege

Listing: Datalader des Verwaltungsprogramms

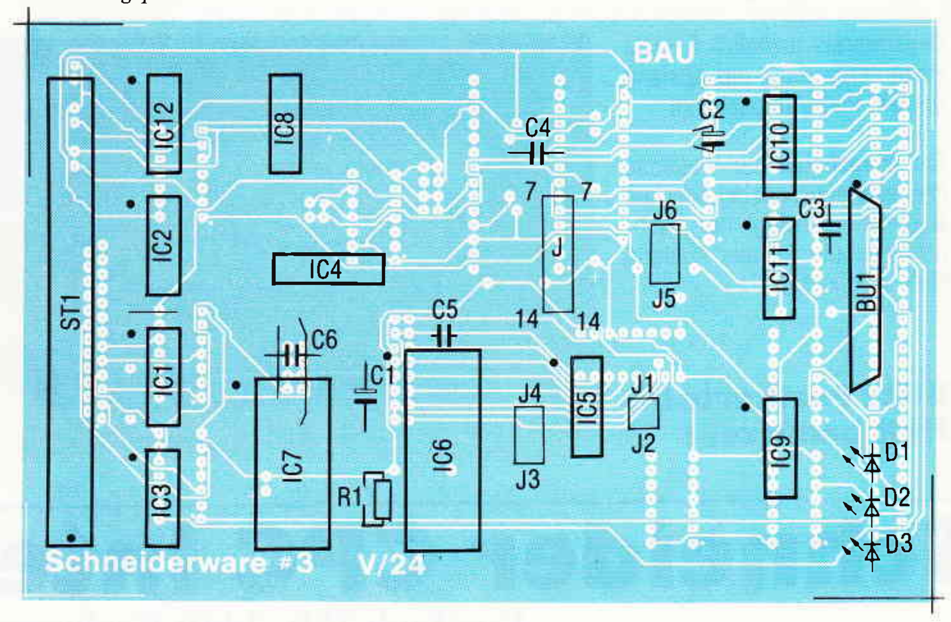
```
10 DATA &01,&0D,&90,&21,&09,&90,&C3,&D1,&BC,&09,  945
20 DATA &90,&0D,&90,&21,&90,&C3,&3F,&90,&C3,&41,  1140
30 DATA &90,&C3,&7C,&90,&C3,&ED,&90,&C3,&91,&91,  1668
40 DATA &C3,&4A,&91,&49,&4E,&49,&54,&B4,&42,&41,  1033
50 DATA &55,&C4,&53,&49,&4F,&49,&4E,&49,&D4,&53,  1035
60 DATA &45,&4E,&C4,&52,&45,&43,&D6,&50,&52,&49,  1010
70 DATA &4E,&D4,&00,&37,&C9,&FE,&02,&C0,&00,&01,  995
80 DATA &EF,&F9,&FD,&21,&5F,&90,&FD,&7E,&00,&ED,  1629
90 DATA &79,&FD,&7E,&01,&ED,&79,&FD,&7E,&02,&ED,  1477
100 DATA &79,&CD,&62,&90,&C9,&36,&76,&B6,&00,&01,  1124
110 DATA &EC,&F9,&DD,&7E,&00,&ED,&79,&DD,&7E,&01,  1538
120 DATA &ED,&79,&03,&DD,&7E,&02,&ED,&79,&DD,&7E,  1415
130 DATA &03,&ED,&79,&C9,&FE,&04,&C0,&01,&E1,&F9,  1487
140 DATA &3E,&00,&ED,&79,&ED,&79,&ED,&79,&3E,&40,  1262
150 DATA &ED,&79,&21,&EC,&90,&AF,&77,&DD,&7E,&00,  1412
160 DATA &FE,&01,&28,&3A,&CB,&AE,&DD,&7E,&02,&FE,  1333
170 DATA &01,&28,&35,&CB,&F6,&CB,&FE,&DD,&7E,&04,  1351
180 DATA &FE,&01,&28,&30,&CB,&A6,&DD,&7E,&06,&FE,  1319
190 DATA &07,&28,&2B,&CB,&86,&CB,&CE,&CB,&D6,&CB,  1456
200 DATA &DE,&01,&E1,&F9,&7E,&ED,&79,&3E,&37,&ED,  1535
210 DATA &79,&CD,&0B,&91,&00,&CD,&3C,&91,&CD,&42,  1163
220 DATA &91,&C9,&CB,&EE,&18,&C4,&CB,&F6,&CB,&BE,  1849
230 DATA &18,&C9,&CB,&E6,&18,&CE,&CB,&86,&CB,&CE,  1634
240 DATA &CB,&96,&CB,&DE,&18,&D3,&CE,&DD,&7E,&00,  1566
250 DATA &F5,&00,&00,&00,&CD,&00,&91,&30,&F8,&F1,  1132
260 DATA &01,&E0,&F9,&ED,&79,&C9,&01,&E1,&F9,&ED,  1745
270 DATA &78,&E6,&01,&C8,&37,&9F,&C9,&21,&00,&60,  1095
280 DATA &22,&1E,&91,&22,&20,&91,&21,&00,&70,&22,  599
290 DATA &22,&91,&22,&24,&91,&C9,&01,&60,&01,&60,  789
300 DATA &00,&70,&00,&70,&3A,&71,&01,&60,&00,&00,  492
310 DATA &00,&00,&00,&81,&01,&92,&00,&2A,&91,&2C,  507
320 DATA &91,&00,&80,&49,&91,&00,&21,&33,&91,&C3,  915
330 DATA &E3,&BC,&21,&2A,&91,&C3,&E3,&BC,&C9,&C9,  1647
340 DATA &E5,&D5,&C5,&F5,&DD,&7E,&00,&CD,&59,&91,  1670
350 DATA &F1,&C1,&D1,&E1,&C9,&2A,&24,&91,&CD,&79,  1618
360 DATA &91,&ED,&4B,&22,&91,&37,&3F,&ED,&42,&CA,  1259
370 DATA &77,&91,&2A,&24,&91,&77,&CD,&79,&91,&22,  1111
380 DATA &24,&91,&37,&3F,&C9,&37,&C9,&C5,&23,&22,  1022
390 DATA &26,&91,&01,&FF,&7F,&37,&3F,&ED,&42,&20,  1019
400 DATA &05,&21,&00,&70,&C1,&C9,&2A,&26,&91,&C1,  962
410 DATA &C9,&CD,&9C,&91,&DD,&66,&01,&DD,&6E,&00,  1362
420 DATA &77,&C9,&CD,&CC,&91,&DA,&B2,&91,&E6,&7F,  1772
430 DATA &21,&B5,&91,&BE,&CA,&B2,&91,&CB,&7E,&C0,  1595
440 DATA &23,&C3,&A7,&91,&3E,&00,&C9,&01,&02,&03,  811
450 DATA &04,&05,&06,&0E,&0F,&10,&11,&12,&13,&14,  134
460 DATA &15,&16,&17,&18,&19,&1A,&1C,&1D,&1F,&80,  357
470 DATA &2A,&1E,&91,&ED,&4B,&20,&91,&37,&3F,&ED,  1061
480 DATA &42,&CA,&E7,&91,&2A,&1E,&91,&7E,&CD,&E9,  1425
490 DATA &91,&22,&1E,&91,&37,&3F,&C9,&37,&C9,&C5,  1126
500 DATA &23,&22,&28,&91,&01,&FF,&6F,&37,&3F,&ED,  976
510 DATA &42,&20,&05,&21,&00,&60,&C1,&C9,&2A,&28,  708
520 DATA &91,&C1,&C9,&F5,&E5,&C5,&D5,&CD,&10,&92,  1790
530 DATA &CD,&1C,&92,&D1,&C1,&E1,&F1,&C9,&CD,&2E,  1699
540 DATA &BD,&D8,&CD,&2D,&92,&D8,&CD,&31,&BD,&C9,  1661
550 DATA &CD,&6A,&92,&E6,&02,&C8,&C5,&01,&E0,&F9,  1560
560 DATA &ED,&78,&C1,&CD,&4A,&92,&C9,&2A,&22,&91,  1397
570 DATA &ED,&4B,&24,&91,&37,&3F,&ED,&42,&CA,&48,  1188
580 DATA &92,&2A,&22,&91,&7E,&CD,&79,&91,&22,&22,  1032
590 DATA &91,&37,&3F,&C9,&37,&C9,&2A,&20,&91,&CD,  1144
600 DATA &E9,&91,&ED,&4B,&1E,&91,&37,&3F,&ED,&42,  1286
610 DATA &CA,&68,&92,&2A,&20,&91,&77,&CD,&E9,&91,  1373
620 DATA &22,&20,&91,&37,&3F,&C9,&37,&C9,&C5,&01,  984
630 DATA &E1,&F9,&ED,&78,&C1,&C9,&00,&00,&00,&00,  1225
640 DATA &00,&00,&00,&00,&00,&00,&00,&00,&00,&00,  0
650 DATA &00, 0
660 dat=0:sz=0:dz= 10
670 FOR adr = &9000 TO  &9280
680 READ byte : dat=dat+1
690   sz=sz+byte
700   POKE adr,byte
710   IF dat < 10 AND adr <  &9280 THEN 750
720  READ chksum
730  IF chksum<>sz THEN PRINT "Fehler in zeile :";dz
740   dz=dz + 10 : sz=0:dat=0
750 NEXT adr
760 MODE 2:PRINT:PRINT"      MAIL.BIN wird gespeichert"
770 PRINT"      Datentraeger fertigmachen, dann TASTE":C
ALL &BB06
780 SAVE"mail.bin",b,&9000,&280
790 END
```

Die Software krönt das Ganze

Nach dem erfolgreichen Aufbau der neuen Karte möchten Sie auch damit arbeiten. Dazu benötigen wir eine geeignete Treibersoftware. Bei deren Erstellung habe ich an diejenigen unter Ihnen gedacht, die weniger Erfahrung in der Assemblerprogrammierung besitzen. Deshalb erstellte ich ein Programm, das es Ihnen ermöglicht, auf einfache Weise durch logische Basicbefehle die Betriebsbedingungen der Karte einzustellen.

Ein solches Programm ausschließlich in Basic zu schreiben, wäre viel zu langsam, und könnte die Geschwindigkeit der seriellen Verarbeitung nicht mithalten. Das Verwaltungsprogramm (Listing 1) laden Sie mittels Datalader in Ihren Rechner und starten es mit Run. Nun werden die Hexadezimalzahlen an die richtige Stelle im Speicher 'gepoked'. Die Startadresse des Programms ist HEX (&) 9000. Nach einem Call %9000 stehen Ihnen folgende neuen Befehle zur Verfügung: Mit **I**BAUD, A, B legen Sie die Baudrate fest. Zuerst wird allerdings der Timer initialisiert. 'A' steht für die Sende- und 'B' für die Empfangsbaudrate. In unserem Fall immer beide gleich einstellen. Der Befehl **I**SIOINIT,A,B,C,D initialisiert den seriellen Baustein, und legt die Übertragungsbedingungen fest. 'A' kann sein 7 (7 BITS) oder 8 (8 BITS),'B' schaltet die Paritypriifung ein (1) oder aus (0), 'C' bedeutet 1 (1) oder 2 (2) Stopbits und 'D' legt fest, auf welchen Wert bei Parity ein getestet werden soll. EVEN (1) bedeutet gerade Anzahl von Einsen oder gerade Parität und ODD (0) heißt, daß auf eine ungerade Anzahl von Einsen getestet werden soll, oder ungerade Parität. Der Befehl **I**SEND,ASC("A") übergibt der seriellen Schnittstelle den Hex-Wert des Zeichens "A", also den Wert Hex &41. Um die Schnittstelle nun bequem von Basic aus bedienen zu können, reservierte ich zwei Speicherbereiche von je 4KB (4096 Zeichen). Diese Buffer werden von einem Programmteil benutzt, der die eigentlichen Druck- und Empfangsroutinen enthält. Dieser Programmteil wird 300 mal in einer Sekunde durch eine Routine des Betriebssystems (Kernel) verwaltet. Dieser Teil erledigt seine Aufgabe geradezu vorbildlich und verwaltet diese Bufferbereiche. Die Anwender, in unserem Fall SIE, übergeben nur noch Werte, oder holen sich die Zeichen mittels unseren 'langsamen' Basicbefehlen und können die Er-

Bestückungsplan



gebnisse gemütlich zur Anzeige oder zum Ausdruck bringen. Die dafür nötigen Befehle lauten IRECV, @A; die Integervariable a enthält das empfangene Zeichen, vorher definieren mit 'Defint a'; und der Befehl IPRINT, ASC ("A") übergibt an den Druckerbuffer den Wert des ASCII-Zeichens "A", das ist die Zahl &41.

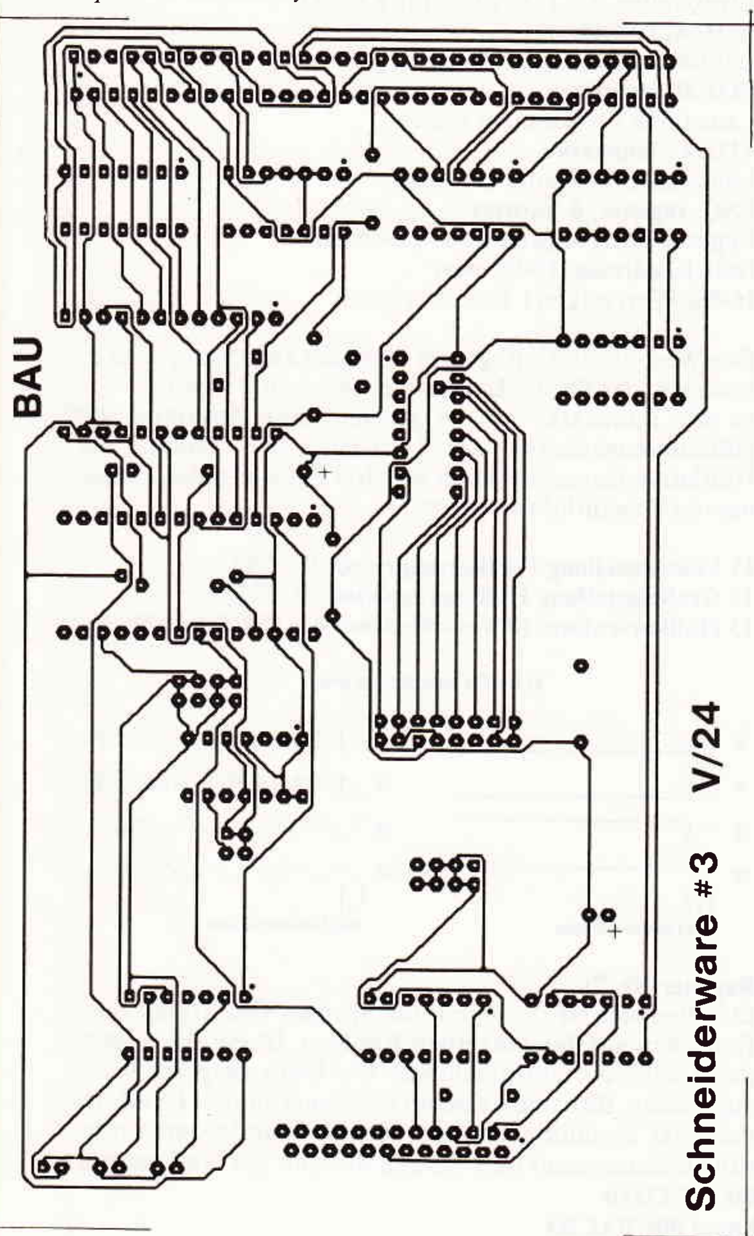
Zu guter LETZT

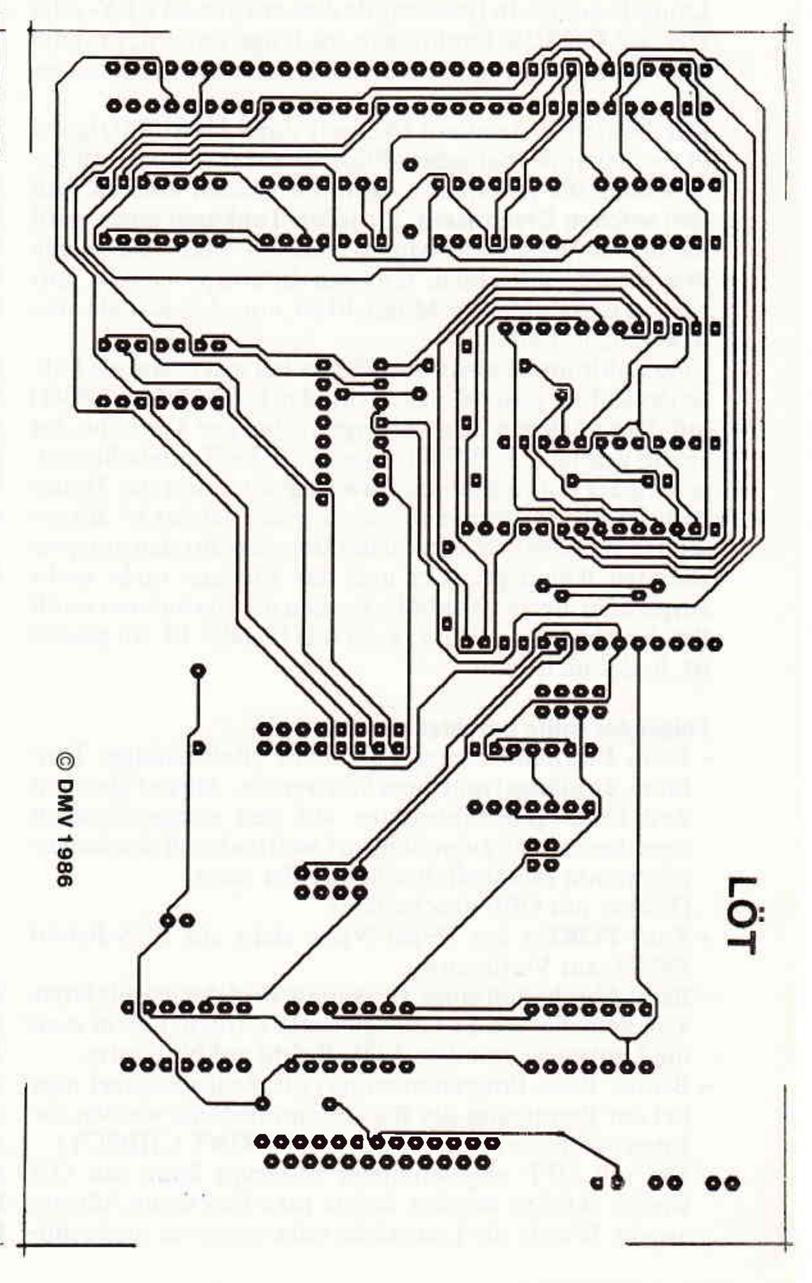
Da die V/24-Schnittstelle ungewohnte Pegel verwendet, mußten besondere Treiberbausteine eingebaut werden. Demnach entspricht ein logisch HIGH bei V/24 einer Spannung von -3 bis -12 Volt und einem logisch LOW einem Pegel von +3 bis +12 Volt. Um diese Pegel zu erzeugen, gibt es mehrere Möglichkeiten. Zum einen stellt die Industrie Schaltkreise her (z.B. MAX 232), die diese Spannungspegel aus einer Spannung von 5 Volt intern erzeugen. Der Nachteil ist momentan die schlechte Bezugsmöglichkeit, da diese Bausteine relativ neu auf dem Markt und zudem teurer sind. Wir verwenden die alteingesessenen V/24-Treiberbausteine (MC 1488 und MC 1489) und führen nach altem Brauch unsere +/-12 Volt an die Karte. Diese Spannungen werden wir aus dem Universalnetzteil entnehmen, das ich Ihnen im nächsten Heft vorstellen möchte. Dieses Netzteil ist sowieso bald vonnöten, da das Netzteil des CPC in absehbarer Zeit ja doch zu schwach wird und wir in einen der nächsten Heften unseren intelligenten Eprommer vorstellen werden, der eine dieser Spannungen benötigt. Vorläufig können Sie die +/-12 V-Spannungen auch durch zwei 9V-Batterien ersetzen; die Schnittstelle arbeitet auch dann noch einwandfrei. Bis dahin wünsche ich Ihnen viel Erfolg beim 'SCHNEIDERN' Ihrer Erweiterungskarten.

(P. Richter/ME)

Stückliste der V/24:
IC 1: 74LS30
IC 2: 74LS30
IC 3: 74LS32
IC 4: 74LS138
IC 5: 74LS74
IC 6: 8251
IC 7: 8253
IC 8: 74LS00
IC 9: 74LS04
IC 10: MC 1488
IC 11: MC 1489
IC 12: 74LS00
R1: 1KOHM
C1: ELKO 1uF
C2: ELKO 100 uF
C3 – C6: 100 nF
D1: LED rot
D2: LED gelb
D3: LED grün
Bu1: SUB-D-Buchse 25-polig
St1: VG-Messerleiste 64-polig a+c

Repro 1+2: Platinenlayout





Interrupt gesteuerte Musik

CPC TIP

für 464-664-6128



In Basic werden Töne mit Interrupts, die jede 1/50 Sekunde aufgerufen werden, programmiert. Allerdings gibt es in den Tonwarteschlangen nur Platz für fünf Töne pro Kanal. Sollen mehr als fünf Töne erzeugt werden, dann wartet der Basic-Interpreter in der entsprechenden Zeile, bis wieder Platz frei ist. Will man dies umgehen, dann muß man ein Programm haben, das automatisch den nächsten Ton in die Register des Tongenerators schreibt und dessen Länge beachtet. In Basic würde dies mit den EVERY- oder ON SQ GOSUB-Funktionen zu lange dauern, deshalb muß das Steuerprogramm in Maschinensprache geschrieben sein.
Der Betriebssystemkern (Kernel) durchläuft fünfzigmal in der Sekunde (bei jedem Bildaufbau) eine Liste von Ereignissen, die jedesmal aufgerufen werden müssen. Mit drei solchen Ereignissen, für jeden Tonkanal eines, wird die interruptgesteuerte Musik realisiert. Man könnte alle drei Kanäle mit einem einzigen Interrupt steuern, nur gäbe es dann nicht die Möglichkeit, einzelne Kanäle aus- bzw. einzuschalten.
Eine Zählroutine zieht vom Zähler bei jedem Aufruf 1 ab. Ist der Zähler gleich 0, dann wird der Tongenerator (PSG) mit dem nächsten Ton versorgt. Ist bei der Tonhöhe das 16. Bit gesetzt (z.B. 32768), dann wird der Tontabellenzeiger wieder auf den Ausgangswert gesetzt und die Tontabelle wieder von vorne durchgearbeitet. Ist das 15. Bit gesetzt (z.B. 16384), dann wird der Interrupt für den entsprechenden Kanal gelöscht und das Ereignis nicht mehr aufgerufen. Bei der Tonhöhe werden nur die unteren zwölf Bits beachtet, eine Prüfung, ob das 13. oder 14. Bit gesetzt ist, findet nicht statt.

Folgendes sollte beachtet werden:

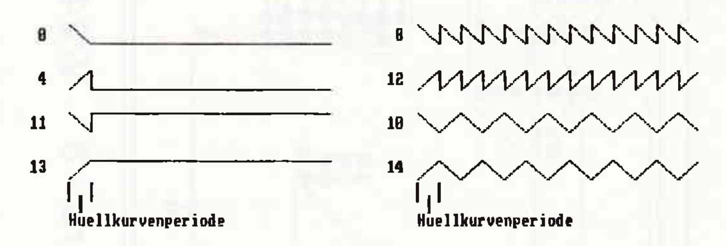
- Beim Einrichten einer Tontabelle (Reihenfolge: Tonhöhe, Tonlänge) muß beachtet werden, daß zur gleichen Zeit keine Interruptroutine auf den entsprechenden Speicherbereich zugreifen darf, weil dadurch das Steuerprogramm eventuell durcheinander gerät. (Vorher mit OFF abschalten!)
- Zum POKEn der 16-Bit-Werte steht ein RSX-Befehl DOKE zur Verfügung.
- Beim Abschalten eines Tonkanals wird der jeweils letzte Ton beibehalten. Er kann gelöscht werden, indem man die Lautstärke mit dem VOL-Befehl auf Null setzt.
- Bei der Basic-Programmierung (z.B. Zeileneingabe) oder bei der Benutzung der Basic-Soundbefehle werden die Interrupt-Töne zerstört (auch bei PRINT CHR$(7)).
- Ein mit OFF abgeschalteter Interrupt kann mit ON wieder aktiviert werden, indem man hier keine Adresse angibt. Wurde die Lautstärke oder sonst ein maßgebliches Register zerstört, dann muß es vorher wieder auf den richtigen Wert gebracht werden.
  War vorher kein Interrupt vorhanden, oder wurde der Interruptblock verändert, so kann dies unvorhersehbare Folgen haben.
- Damit ein Ton erklingen kann, muß die Lautstärke des Kanals mindestens 1, das Register Nr. 7 richtig geladen und eine Tontabelle vorhanden sein.
- Die Tonlänge wird in 1/50 Sekunden angegeben, die Tonhöhe wie im Handbuch in Anhang 7 (Seite 1 ff.) beschrieben.

Für die Programmierung stehen RSX-Befehle zur Verfügung:

OFF
Abschalten aller drei Interrupts
OFF.A
Abschalten des Interrupts für Kanal A
OFF.B
Abschalten des Interrupts für Kanal B
OFF.C
Abschalten des Interrupts für Kanal C
ON.A, adresse
Einschalten des Interrupts für Kanal A
ON.B, adresse
Einschalten des Interrupts für Kanal B
ON.C, adresse
Einschalten des Interrupts für Kanal C
VOL.A, lautstärke
Lautstärke für Kanal A setzen
VOL.B, lautstärke
Lautstärke für Kanal B setzen
VOL.C, lautstärke
Lautstärke für Kanal C setzen
PSG, register, 8-bit-wert
Register des Tongenerators beschreiben
DOKE, adresse, 16-bit-wert
16-Bit-Wert poken (Intel-Format)

Der Wert des PSG-Registers darf sich zwischen 0 und 13 bewegen, der für die Lautstärke zwischen 0 und 16.
Ist die Lautstärke auf 16 gesetzt, dann bestimmt der Hüllkurvengenerator die Lautstärke des Tones. Der Hüllkurvengenerator kann mit drei Registern des Tongenerators beeinflußt werden:

11 Feineinstellung Hüllkurvenperiode 0 - 255
12 Grobeinstellung Hüllkurvenperiode 0 - 255
13 Hüllkurvenform (8 Verschiedene) 0 - 15 (siehe Abb. 1)



Register Nr. 7:
Das Register Nr. 7 steuert das Sperren von Tönen oder Rauschen auf den einzelnen Kanälen. Ist ein Bit gesetzt, dann wird die entsprechende Funktion gesperrt, sonst zugelassen. Bit 6 und 7 sind für die Steuerung der I/O-Ports des PSG zuständig. Sie sind für die Soundprogrammierung uninteressant und werden deshalb auf Null gesetzt.
Bit 76543210
Kanal 00CBACBA

Die Bits 0 - 2 steuern die Töne, die Bits 3 - 5 das Rauschen. (Hermann Röscheisen)

Der Assembler-Quellcode befindet sich aus technischen Gründen auf der zugehörigen CPC-Databox.

Einsprung-Referenz:

Für die Anpassung des Programms an CPC 664 und CPC 6128 hier die Einsprünge in die Firmware und ins Basic-ROM:

B900 KL UROM ENABLE
oberes ROM öffnen
BCD1 KL LOG EXT
RSX einbinden
BCD7 KL NEW FRAME FLY
neuen Frame Fly Interrupt initialisieren und einbinden
BCDD KL DEL FRAME FLY
Frame Fly Interrupt löschen
BD34 MC SOUND REGISTER
Sound-Register laden
CA94 BAS ERROR
Basic-Fehlermeldung ausgeben

```
1000 REM                                        [272]
1010 REM Interrupt mastered music               [2661]
1020 REM Copyright 5/1986 by H.R.               [2297]
1030 REM                                        [272]
1040 a=&A400:e=&A627:z=1150                     [1682]
1050 MODE 2:PRINT"Bitte warten !"               [2295]
1060 MEMORY a-1:FOR n=a TO e:READ d$:IF LE      [2105]
FT$(d$,1)<>"&"THEN 1090
1070 IF c<>VAL(d$)THEN CLS:PRINT"Checksumm      [5293]
enfehler in Zeile";z:END
1080 c=0:d$="":IF n=e THEN 1120:ELSE n=n-1      [3278]
:z=z+10:GOTO 1110
1090 d$="&"+d$                                  [308]
1100 POKE n,VAL(d$):c=c+VAL(d$)                 [1925]
1110 IF n<e THEN NEXT                           [1224]
1120 CLS:PRINT"Format: SAVE"CHR$(34)"IMM.B      [3680]
IN"CHR$(34)",B,&A400,&227"
1130 PRINT:PRINT"RSX-Einbindung mit CALL &      [3251]
A400."
1140 END                                        [110]
1150 DATA 01,0F,A4,21,53,A6,CD,D1,&36C          [2070]

1160 DATA BC,3E,C9,32,00,A4,C9,35,&397          [962]
1170 DATA A4,C3,6A,A4,C3,83,A4,C3,&522          [565]
1180 DATA 9C,A4,C3,B5,A4,C3,B0,A5,&574          [711]
1190 DATA C3,E8,A5,C3,20,A6,C3,BE,&55A          [1989]
1200 DATA A4,C3,C2,A4,C3,C6,A4,C3,&5BD          [2024]
1210 DATA DA,A4,C3,FB,A4,4F,4E,2E,&4AB          [2622]
1220 DATA C1,4F,4E,2E,C2,4F,4E,2E,&319          [2056]
1230 DATA C3,4F,46,C6,4F,46,46,2E,&327          [1377]
1240 DATA C1,4F,46,46,2E,C2,4F,46,&321          [1996]
1250 DATA 46,2E,C3,56,4F,4C,2E,C1,&317          [2148]
1260 DATA 56,4F,4C,2E,C2,56,4F,4C,&2D2          [1349]
1270 DATA 2E,C3,50,53,C7,44,4F,4B,&339          [2103]
1280 DATA C5,00,B7,CA,25,A5,CD,10,&3ED          [1502]
1290 DATA A5,22,26,A6,DD,66,01,DD,&3B4          [1141]
1300 DATA 6E,00,22,2C,A6,22,32,A6,&25C          [1992]
1310 DATA C3,25,A5,B7,CA,36,A5,CD,&4B6          [3095]
1320 DATA 10,A5,22,28,A6,DD,66,01,&2E9          [1956]
1330 DATA DD,6E,00,22,2E,A6,22,34,&297          [911]
1340 DATA A6,C3,36,A5,B7,CA,47,A5,&4B1          [1945]
1350 DATA CD,10,A5,22,2A,A6,DD,66,&3B7          [1697]
1360 DATA 01,DD,6E,00,22,30,A6,22,&266          [1325]
1370 DATA 36,A6,C3,47,A5,CD,B0,A5,&4AD          [1207]
1380 DATA CD,E8,A5,C3,20,A6,06,08,&3F1          [1909]
1390 DATA 18,06,06,09,18,02,06,0A,&057          [605]
1400 DATA FE,01,C2,19,A5,DD,7E,00,&3DA          [829]
1410 DATA FE,11,D2,1D,A5,4F,78,C3,&42D          [1416]
1420 DATA 34,BD,FE,02,C2,19,A5,DD,&44E          [1232]
1430 DATA 7E,01,B7,C2,1D,A5,DD,7E,&415          [1516]
1440 DATA 03,B7,C2,1D,A5,DD,7E,02,&39B          [1466]
1450 DATA FE,0E,D2,1D,A5,DD,4E,00,&3CB          [2266]
1460 DATA C3,34,BD,FE,02,C2,19,A5,&434          [950]
1470 DATA DD,66,03,DD,6E,02,DD,56,&3C6          [1916]
1480 DATA 01,DD,5E,00,73,23,72,C9,&30D          [1816]
1490 DATA FE,01,C2,19,A5,21,00,00,&2A0          [951]
1500 DATA C9,1E,02,18,02,1E,05,CD,&1F3          [2119]
1510 DATA 00,B9,C3,94,CA,21,38,A6,&3D9          [1416]
1520 DATA E5,CD,DD,BC,E1,11,58,A5,&53A          [1505]
1530 DATA 01,00,81,C3,D7,BC,21,41,&33A          [1531]
1540 DATA A6,E5,CD,DD,BC,E1,11,65,&548          [1355]
1550 DATA A5,01,00,81,C3,D7,BC,21,&39E          [1845]
1560 DATA 4A,A6,E5,CD,DD,BC,E1,11,&52D          [931]
1570 DATA 72,A5,01,00,81,C3,D7,BC,&3EF          [1232]
1580 DATA 2A,26,A6,7C,B5,CA,7F,A5,&415          [1581]
1590 DATA 2B,22,26,A6,C9,2A,28,A6,&2DA          [1724]
1600 DATA 7C,B5,CA,B6,A5,2B,22,28,&3CB          [1901]
1610 DATA A6,C9,2A,2A,A6,7C,B5,CA,&464          [1881]
1620 DATA EE,A5,2B,22,2A,A6,C9,2A,&3A3          [1225]
1630 DATA 2C,A6,7E,23,4E,CB,79,C2,&3C7          [1792]
1640 DATA A7,A5,CB,71,C2,B0,A5,23,&4C2          [2172]
1650 DATA 5E,23,56,23,22,2C,A6,67,&255          [1589]
1660 DATA 3E,01,CD,34,BD,AF,4C,CD,&3C5          [2410]
1670 DATA 34,BD,EB,22,26,A6,C9,2A,&3BD          [2032]
1680 DATA 32,A6,22,2C,A6,C3,7F,A5,&3B3          [1539]
1690 DATA 21,38,A6,C3,DD,BC,2A,2E,&3B3          [1194]
1700 DATA A6,7E,23,4E,CB,79,C2,DF,&47A          [1251]
1710 DATA A5,CB,71,C2,E8,A5,23,5E,&4B1          [1444]
1720 DATA 23,56,23,22,2E,A6,67,3E,&237          [1379]
1730 DATA 03,CD,34,BD,3E,02,4C,CD,&31A          [1839]
1740 DATA 34,BD,EB,22,28,A6,C9,2A,&3BF          [2369]
1750 DATA 34,A6,22,2E,A6,C3,B6,A5,&3EE          [1258]
1760 DATA 21,41,A6,C3,DD,BC,2A,30,&3BE          [1346]
1770 DATA A6,7E,23,4E,CB,79,C2,17,&3B2          [917]
1780 DATA A6,CB,71,C2,20,A6,23,5E,&3EB          [848]
1790 DATA 23,56,23,22,30,A6,67,3E,&239          [1216]
1800 DATA 05,CD,34,BD,3E,04,4C,CD,&31E          [1741]
1810 DATA 34,BD,EB,22,2A,A6,C9,2A,&3C1          [1640]
1820 DATA 36,A6,22,30,A6,C3,EE,A5,&42A          [846]
1830 DATA 21,4A,A6,C3,DD,BC,00,00,&36D          [1643]
```

```
100 REM                                         [272]
110 REM Demonstration 1: Menuett                [2245]
120 REM Copyright 5/1986 by H.R.                [2297]
130 REM                                         [272]
140 MEMORY &9FFF:LOAD"IMM.BIN",&A400            [867]
150 CALL &A400:|OFF:|PSG,7,63                   [2493]
160 RESTORE 260:adr=&A000:GOSUB 230             [1627]
170 RESTORE 320:adr=&A100:GOSUB 230             [1633]
180 RESTORE 370:adr=&A200:GOSUB 230             [2242]
190 |PSG,7,&X111000                             [1085]
200 |ON.A,&A000:|ON.B,&A100:|ON.C,&A200         [2871]
210 |VOL.A,10:|VOL.B,10:|VOL.C,10               [1250]
220 END                                         [110]
230 READ a:IF a=-1 THEN RETURN                  [1393]
240 |DOKE,adr,a:adr=adr+2:GOTO 230              [2921]
250 REM  Kanal A (ab Adresse &A000)             [1538]
260 DATA 190,10,213,10,239,20,284,20            [2044]
270 DATA 213,20,179,20,190,20,213,10            [2278]
280 DATA 239,10,253,20,190,10,213,10            [1543]
290 DATA 239,20,253,10,284,10,301,30            [839]
300 DATA 284,10,284,60,32768,-1                 [1477]
310 REM  Kanal B (ab Adresse &A100)             [2051]
320 DATA 0,40,568,10,638,10,716,20              [1760]
330 DATA 851,20,758,20,676,20,602,20            [1821]
340 DATA 758,20,568,20,426,20,379,40            [1148]
350 DATA 0,65,32768,-1                          [1140]
360 REM  Kanal C (ab Adresse &A200)             [1427]
370 DATA 0,240,758,20,568,40,1136,20            [1207]
380 DATA 32768,-1                               [603]
```

```
100 REM                                         [272]
110 REM Demonstration 2: Tonleiter              [2403]
120 REM  Copyright 5/1986 by H.R.               [2297]
130 REM                                         [272]
140 MEMORY &9FFF:LOAD"IMM.BIN",&A400            [867]
150 CALL &A400:|OFF:|PSG,7,63                   [2493]
160 adr=&A000:FOR oct=0 TO 1:GOSUB 260:NEX      [2677]
T
170 |DOKE,adr,32768                             [1751]
180 adr=&A100:FOR oct=1 TO 2:GOSUB 260:NEX      [2017]
T
190 |DOKE,adr,32768                             [1751]
200 adr=&A200:FOR oct=-1 TO 0:GOSUB 260:NE      [3594]
XT
210 |DOKE,adr,32768                             [1751]
220 |VOL.A,15:|VOL.B,15:|VOL.C,15               [1204]
230 |PSG,7,&X111000                             [1085]
240 |ON.A,&A000:|ON.B,&A100:|ON.C,&A200         [2871]
250 END                                         [110]
260 FOR tone=1 TO 12                            [1028]
270 freq=440*(2^(oct+(tone-10)/12))             [1786]
280 period=ROUND(62500/freq)                    [1689]
290 |DOKE,adr,period:adr=adr+2                  [3400]
300 |DOKE,adr,10:adr=adr+2:NEXT                 [2983]
310 RETURN                                      [555]
```

Lister II

für CPC, Joyce CP/M

Für das Ausdrucken von Textdateien unter CP/M hat man ursprünglich zwei Möglichkeiten: Durch Drücken von CTRL-P wird alles, was auf dem Bildschirm ausgegeben wird, auch auf dem Drucker protokolliert. Mit **TYPE Dateiname** wird dann die Datei auf Bildschirm und Drucker ausgegeben. Allerdings werden dabei auch alle Systemmeldungen auf Papier gedruckt und stören das Erscheinungsbild des Textes. Ein erneutes Drücken von CTRL-P schaltet diesen Modus wieder ab.

Als Alternative kann der Text mit Hilfe des PIP-Dienstprogrammes (PIP lst:=dateiname[optionen]) zu Papier gebracht werden. Hierbei stehen vielfältige Optionen zur Manipulation des auszudruckenden Textes zur Verfügung, z.B. kann die Seitenlänge bestimmt werden oder der Text mit Zeilennummern versehen werden.

Allerdings besteht bei beiden keine einfache Möglichkeit, die Schriftart, mit der gedruckt werden soll, festzulegen. Dazu müßte man den Drucker entweder von Basic aus einstellen oder auf das Dienstprogramm SETLST zurückgreifen. Um diesem Umstand zu entgehen, wurde Lister II geschrieben.

Das Programm ist vollständig in Turbo-Pascal 3.00 geschrieben und wurde auf einem CPC 6128 entwickelt. Um das Programm in der CPC 464/664 Turbo-Version compilieren zu können, muß die Frage "Include Error Messages?" mit "Nein" beantwortet werden. Das Programm ist ebenfalls auf dem Joyce lauffähig. Bei den Steuercodes für die Aktivierung/Abschaltung der einzelnen Druckerfunktionen im Drucker-Menue handelt es sich um Codes für den Drucker **Brother M-1009**, und müssen bei Verwendung eines anderen Druckers eventuell entsprechend geändert werden.

Das Programm kann nicht durch Drücken von CTRL-C abgebrochen werden; wem dies jedoch nicht liegt, ersetzt die Compiler-Option {$C-} mit {$C+} im Quelltext.

Da das Programm vollständig menuegesteuert ist, braucht man nur die entsprechende Funktion zu wählen und das Programm führt diese sofort aus. Das danach ertönende Beep-Signal bestätigt die Abarbeitung der Funktion.

Lister II besteht aus drei Teilen:

- Das Drucker-Menue übernimmt die Einstellung der zu druckenden Schriftart, wobei auch Kombinationen verschiedener Arten möglich sind (näheres dem Drucker-Handbuch entnehmen!). Hier hat man auch die Möglichkeit, eine Überschrift zu drucken.
- Der Punkt 2 im Hauptmenue ermöglicht dann den Ausdruck einer Textdatei in der vorher eingestellten Schriftart. Dabei wird der gedruckte Text auch auf dem Bildschirm angezeigt.
- Mit Punkt 3 kann man einen einzugebenden Text drukken lassen, um die gewählte Schriftart zu kontrollieren.

Durch Drücken der X-Taste wird das Programm verlassen und zu CP/M zurückgekehrt. (Stefan Schulte Strathaus)

```
Program ListerII;
{$C+}

(*****************************************************************************
*                                                                           *
*           Source Lister Version II with Printer Set-up                    *
*       Copyright (c) 1986 by The Red Rock Software Development             *
*                                                                           *
*****************************************************************************

 Application : All Systems using Turbo Pascal with brother Printer or
                                                            compatibles.)

(** You can't stop this Programm with CTRL-C **)


Procedure BELL;
begin
  write (char (7));
end;


Procedure MainMenu;
Var  Choose, Menu :Char;
              A,i :Integer;
         Headline :String [255];
         Filename :String [15];
           FilVar :Text;
              TXT :Char;
```

```
end;
GotoXY (19,3); write (char(134));
GotoXY (55,3); write (char(140));
GotoXY (19,4); write (char(133),'     >>>>> DRUCKER  Menu <<<<<      ',char(133));
LowVideo;
GotoXY (20,6); write ('[0]........Kursivschrift einstellen');
GotoXY (20,7); write ('[1].........Kursivschrift abstellen');
GotoXY (20,8); write ('[2]...Ueberschrift Gross und Kursiv');
GotoXY (20,09); write ('[3]..........Ueberschrift nur Gross');
GotoXY (20,10); write ('[4].Kondensierte Schrift einstellen');
GotoXY (20,11); write ('[5]........Normalschrift einstellen');
GotoXY (20,12); write ('[6].......Doppelanschlag einstellen');
GotoXY (20,13); write ('[7]........Doppelanschlag abstellen');
GotoXY (20,14); write ('[X]...........Zurueck zum Hauptmenu');
NormVideo;
GotoXY (19,16); write (char(133));
GotoXY (55,16); write (char(133));
For I:=1 to 35 do begin
  GotoXY (19+i,17); write (char(138));
end;
GotoXY (19,17); write (char(131));
GotoXY (55,17); write (char(137));
repeat
  GotoXY (0,0); read (kbd,Menu);
  Menu:=UpCase(Menu);
Until Menu in ['0','1','2','3','4','5','6','7','X'];
Case Menu of
  '0' : begin (Kursivschrift einstellen)
```

```
begin
  ClrScr; LowVideo;
  write ('    LISTER Version II  Copyright 1986 by The Red Rock Software Development    ');
  NormVideo;
  For A:=0 to 80 do begin
    GotoXY (A,3);write ('_');
    GotoXY (A,23);write ('_');
  end;
  LowVideo;
  GotoXY (22,8);write ('[1].........Drucker Menu anwaehlen');
  GotoXY (22,9);write ('[2].........Source Listing drucken');
  GotoXY (22,10);write ('[3]...................Text drucken');
  GotoXY (22,11);write ('[X]................Zurueck zu CP/M');
  NormVideo;
  repeat
    GotoXY (22,13);write ('Bitte waehlen Sie : ');
    read (kbd,Choose);Choose:=UpCase (Choose);
  Until Choose in ['1','2','3','X'];

  Case Choose of
    '3' : begin
          ClrScr;
          writeln ('Bitte den Text eingeben,der gedruckt werden soll : ');
          readln (Headline);
          writeln (lst,Headline);
          MainMenu;
        end;
    '2' : begin
          repeat
            ClrScr;
            write ('Bitte den Dateinamen mit Extension eingeben : ');
            LowVideo;
            read(Filename);
            NormVideo;
            Writeln;
            Assign (FilVar,Filename);
            {$I-}
            Reset (FilVar);
            {$I+}
          Until IOresult=0;
          While not EOF (FilVar) do begin
            read (FilVar,TXT);
            write (lst,TXT); write (TXT);
          end;
          close (FilVar);
          MainMenu;
        end;
    'X' : begin
          ClrScr; HALT;
        end;
     '1' : Begin
           ClrScr;
           For I:=1 to 35 do begin
             GotoXY (19+i,3); write(char(138));
             write (lst,char (27),char(52));
             Bell;
             MainMenu;
           end; {Funktion 0: beendet}
       '1' : begin {Kursivschrift abstellen}
             write (lst,char (27),char(53));
             Bell;
             MainMenu;
           end; {Funktion 1: beendet}
       '2' : begin {Ueberschrift Gross und Kursiv}
             writeln (lst);
             GotoXY (1,23); write ('>'); read (Headline);
             writeln (lst,char(14),char (27),char (52),Headline,char (27),char (53));
             Bell;
             MainMenu;
           end; {Funktion 2: beendet}
       '3' : begin {Ueberschrift nur Gross}
             writeln (lst);
             GotoXY (1,23); write ('>'); read (Headline);
             writeln (lst,char (27),char (87),char(1),Headline,char (27),char (87),char (0));
             Bell;
             MainMenu;
           end; {Funktion 3: beendet}
       '4' : begin {Komprimierte Schrift}
             write (lst,char (15));
             Bell;
             MainMenu;
           end; {Funktion 4: beendet}
       '5' : begin {Normale Schrift}
             write (lst,char (18));
             Bell;
             MainMenu;
           end; {Funktion 5: beendet}
       '6' : begin {Doppelanschlag}
             write (lst,char (27),char (71));
             Bell;
             MainMenu;
           end; {Funktion 6: beendet}
       '7' : begin {Doppelanschlag abstellen}
             write (lst,char (27),char (72));
             Bell;
             MainMenu;
           end; {Funktion 7: beendet}
       'X' : MainMenu;
     end; (* CASE *)
   end; (* '1' *)
  end; (* CASE *)
end; (* mainmanu *)

begin
  mainmenu;
end.
```

CPC-Listingservice für 464-664-6128



Wie in jedem Heft, finden Sie auch hier wieder den Listingservice, der Ihnen helfen soll, die in unserem Magazin abgedruckte "Paperware" in "Software" für Ihren Rechner zu verwandeln. Neben den Piktogrammen für Kassetten-/Diskettenbetrieb und der Angabe der Lauffähigkeit finden Sie die Super-Eintipphilfe "Checksummer". Um den Checksummer benutzen zu können, genügt es, das Programm einmal aufzurufen; es generiert selbständig den Maschinencode, aktiviert den Checksummer, gibt eine Bereitschaftsmeldung aus und löscht sich anschließend selbst wieder. Ab jetzt wird jede Eingabe, nachdem man ENTER gedrückt hat, mit der Ausgabe einer Quersumme quittiert. Beim Eintippen von Programmlistings kann man nun durch den Vergleich der beiden in eckigen Klammern stehenden Zahlen feststellen, ob man sich vertippt hat und ggf. den Fehler sofort berichtigen. Des weiteren hat man die Möglichkeit, mit dem RSX-Befehl CHECK,s,z ein Listing mit den dazugehörigen Checksummen zu erzeugen. Die Parameter s und z definieren den Ausgabekanal (s = 0 bis 9) und die Zeilennummer, ab der mit der Ausgabe begonnen werden soll; wird z nicht angegeben, so wird an dessen Stelle 1 angenommen.

Vor dem Druck sollte man mit WIDTH x die maximale Breite (x) einer Zeile festlegen. Mit Hilfe der RSX-Befehle ON und OFF kann der Cheksummer ein- bzw. ausgeschaltet werden.

Bei der Berechnung der Quersumme werden die Zeilennummer und Leerzeichen überlesen; zwischen Groß- und Kleinschreibung wird nur innerhalb zweier Anführungsstriche unterschieden, da ansonsten die Schreibweise von Befehlsworten und Variablennamen keinen Einfluß auf die Funktionstüchtigkeit eines Basicprogrammes hat.

(Thomas Fippl)



Version 464

```
100 REM *** CHECKSUM 464 ***
110 MEMORY &A4FF
120 FOR a%=&A500 TO &A607
130 READ byte$
140 POKE a%,VAL("&"+byte$)
150 NEXT
160 PRINT
170 PRINT"CHECKSUM v2"
180 CALL &A500:¦ON:NEW
190 :
200 DATA 21,09,a5,01,0d,a5,c3,d1
210 DATA bc,00,00,00,00,18,a5,c3
220 DATA 2a,a5,c3,2f,a5,c3,43,a5
230 DATA 4f,ce,4f,46,c6,43,48,45
240 DATA 43,cb,00,00,cf,98,aa,c3
250 DATA a8,a5,21,27,a5,18,03,21
260 DATA 24,a5,28,06,cd,00,b9,c3
270 DATA 06,dd,11,3a,bd,01,03,00
280 DATA ed,b0,c9,4f,cd,00,b9,0d
290 DATA 28,08,0d,20,ea,dd,7e,02
300 DATA 18,04,7b,11,01,00,cd,a2
310 DATA c1,cd,a3,e7,e5,4e,23,46
320 DATA 23,5e,23,56,e1,78,b1,c8
330 DATA cd,3c,c4,e5,09,e3,cd,63
340 DATA e1,21,a4,ac,cd,7a,a5,e1
350 DATA 18,e2,e5,cd,ba,a5,e3,cd
360 DATA 98,a5,cd,96,f2,e3,cd,f6
370 DATA a5,cd,4e,c3,e1,7e,a7,c8
380 DATA cd,98,a5,cd,4e,c3,18,f5
390 DATA 3a,24,ac,d6,08,47,7e,a7
400 DATA c8,cd,45,e1,23,10,f7,c9
410 DATA cd,24,a5,f5,c5,d5,e5,cd
420 DATA ba,a5,cd,f6,a5,e1,d1,c1
430 DATA f1,c9,eb,1b,af,47,67,6f
440 DATA 2f,32,23,a5,13,1a,d6,30
450 DATA 38,04,fe,0a,38,f6,1a,13
460 DATA a7,c8,4f,3a,23,a5,a1,fe
470 DATA 20,28,f3,79,fe,22,20,07
480 DATA 3a,23,a5,2f,32,23,a5,3a
490 DATA 23,a5,a7,79,c4,8a,ff,4f
500 DATA ad,07,6f,09,18,d8,3e,20
510 DATA cd,5c,c3,3e,5b,cd,5c,c3
520 DATA cd,79,ee,3e,5d,c3,5c,c3
```

Version 664

```
100 REM *** CHECKSUM 664 ***
110 MEMORY &A4FF
120 FOR a%=&A500 TO &A607
130 READ byte$
140 POKE a%,VAL("&"+byte$)
150 NEXT
160 PRINT
170 PRINT"CHECKSUM v2"
180 CALL &A500:¦ON:NEW
190 :
200 DATA 21,09,a5,01,0d,a5,c3,d1
210 DATA bc,00,00,00,00,18,a5,c3
220 DATA 2a,a5,c3,2f,a5,c3,43,a5
230 DATA 4f,ce,4f,46,c6,43,48,45
240 DATA 43,cb,00,00,cf,02,ac,c3
250 DATA a8,a5,21,27,a5,18,03,21
260 DATA 24,a5,28,06,cd,00,b9,c3
270 DATA 4c,cb,11,5b,bd,01,03,00
280 DATA ed,b0,c9,4f,cd,00,b9,0d
290 DATA 28,08,0d,20,ea,dd,7e,02
300 DATA 18,04,7b,11,01,00,cd,a9
310 DATA c1,cd,69,e8,e5,4e,23,46
320 DATA 23,5e,23,56,e1,78,b1,c8
330 DATA cd,75,c4,e5,09,e3,cd,59
340 DATA e2,21,8a,ac,cd,7a,a5,e1
350 DATA 18,e2,e5,cd,ba,a5,e3,cd
360 DATA 98,a5,cd,58,f3,e3,cd,f6
370 DATA a5,cd,9b,c3,e1,7e,a7,c8
380 DATA cd,98,a5,cd,9b,c3,18,f5
390 DATA 3a,09,ac,d6,08,47,7e,a7
400 DATA c8,cd,22,e2,23,10,f7,c9
410 DATA cd,24,a5,f5,c5,d5,e5,cd
420 DATA ba,a5,cd,f6,a5,e1,d1,c1
430 DATA f1,c9,eb,1b,af,47,67,6f
440 DATA 2f,32,23,a5,13,1a,d6,30
450 DATA 38,04,fe,0a,38,f6,1a,13
460 DATA a7,c8,4f,3a,23,a5,a1,fe
470 DATA 20,28,f3,79,fe,22,20,07
480 DATA 3a,23,a5,2f,32,23,a5,3a
490 DATA 23,a5,a7,79,c4,ab,ff,4f
500 DATA ad,07,6f,09,18,d8,3e,20
510 DATA cd,a3,c3,3e,5b,cd,a3,c3
520 DATA cd,49,ef,3e,5d,c3,a3,c3
```

Version 6128

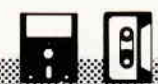
```
100 REM *** CHECKSUM 6128 ***
110 MEMORY &A4FF
120 FOR a%=&A500 TO &A607
130 READ byte$
140 POKE a%,VAL("&"+byte$)
150 NEXT
160 PRINT
170 PRINT"CHECKSUM v2"
180 CALL &A500:¦ON:NEW
190 :
200 DATA 21,09,a5,01,0d,a5,c3,d1
210 DATA bc,00,00,00,00,18,a5,c3
220 DATA 2a,a5,c3,2f,a5,c3,43,a5
230 DATA 4f,ce,4f,46,c6,43,48,45
240 DATA 43,cb,00,00,cf,02,ac,c3
250 DATA a8,a5,21,27,a5,18,03,21
260 DATA 24,a5,28,06,cd,00,b9,c3
270 DATA 49,cb,11,5e,bd,01,03,00
280 DATA ed,b0,c9,4f,cd,00,b9,0d
290 DATA 28,08,0d,20,ea,dd,7e,02
300 DATA 18,04,7b,11,01,00,cd,a6
310 DATA c1,cd,64,e8,e5,4e,23,46
320 DATA 23,5e,23,56,e1,78,b1,c8
330 DATA cd,72,c4,e5,09,e3,cd,54
340 DATA e2,21,8a,ac,cd,7a,a5,e1
350 DATA 18,e2,e5,cd,ba,a5,e3,cd
360 DATA 98,a5,cd,53,f3,e3,cd,f6
370 DATA a5,cd,98,c3,e1,7e,a7,c8
380 DATA cd,98,a5,cd,98,c3,18,f5
390 DATA 3a,09,ac,d6,08,47,7e,a7
400 DATA c8,cd,1d,e2,23,10,f7,c9
410 DATA cd,24,a5,f5,c5,d5,e5,cd
420 DATA ba,a5,cd,f6,a5,e1,d1,c1
430 DATA f1,c9,eb,1b,af,47,67,6f
440 DATA 2f,32,23,a5,13,1a,d6,30
450 DATA 38,04,fe,0a,38,f6,1a,13
460 DATA a7,c8,4f,3a,23,a5,a1,fe
470 DATA 20,28,f3,79,fe,22,20,07
480 DATA 3a,23,a5,2f,32,23,a5,3a
490 DATA 23,a5,a7,79,c4,ab,ff,4f
500 DATA ad,07,6f,09,18,d8,3e,20
510 DATA cd,a0,c3,3e,5b,cd,a0,c3
520 DATA cd,44,ef,3e,5d,c3,a0,c3
```



Data Media Computerclub
Data Media GmbH, Ruhralle 55, 4600 Dortmund
Tel.: (02 31) 12 50 71-3
Bestellkarte benutzen!
MAIL

Rückfahrkarte für Parameter

für 464



Wenn es um Parameter von Grafik, Sound usw. geht, zeigt sich der Rechner (zumindest von Basic aus) nicht sehr auskunftsfreudig: Man kann zwar (fast) alle Parameter mit Basic-Befehlen an den Rechner übergeben und einige wenige auf dieselbe Weise auch wieder zurückerhalten, die meisten jedoch verschwinden auf Nimmerwiedersehen im Rechner.
Glücklicherweise werden sie aber (meistens unverändert) irgendwo im RAM abgelegt, damit das Betriebssystem sie dort bei Bedarf abrufen kann. Doch was dem Betriebssystem recht ist, kann dem User nur billig sein, und so holt er sich eben (mittels PEEK) alle die Parameter wieder zurück, die ihm das Basic sonst vorenthält. Wie, das soll die untenstehende Liste zeigen.
Wenn Sie einmal etwas andere Werte zurückbekommen als Sie kurz zuvor eingegebenen haben, so sollte Sie das nicht wundern: Der Rechner überprüft und korrigiert natürlich alle Werte, bevor er sie im RAM ablegt.

(Thomas Kochmann)

```
                    "Rückfahrkarte für Parameter"

BORDER b(1),b(2)          a(1) = PEEK(&B1D9)
                          a(2) = PEEK(&B1EA)
                          weiter siehe Tabelle 1

CALL a                    a = PEEK(&AE72)+256*PEEK(&AE73)

CLG i                     j = PEEK(&B339)
                          weiter siehe Tabelle 2

DEG                       PEEK(&B8F7)=0   -> RAD
                          PEEK(&B8F7)=255 -> DEG

DI                        PEEK(&B195)=0  -> DI
                          PEEK(&B195)=32 -> EI

DRAW x,y,i                s. PLOT x,y,i

DRAWR x,y,i               s. PLOT x,y,i

EI                        s. DI

ENT n,b(1),b(2),...       b(i) = PEEK(&B6FA+16*ABS(n)+i)

ENV n,b(1),b(2),...       b(i) = PEEK(&B60A+16*n+i)

INK n,b(1),b(2)           a(1) = PEEK(&B1DA+n)
                          a(2) = PEEK(&B1EB+n)
                          weiter siehe Tabelle 1

KEY n,text$               Die Texte werden von Adresse &B446 an im
                          RAM abgelegt. Jeder Text beginnt mit der
                          der Anzahl seiner Zeichen (= LEN(text$)).
                          Die einzelnen Texte werden in der Reihen-
                          folge n=128, n=129, usw. hintereinander
                          abgelegt.

KEY DEF n,x,a,b,c         x = (PEEK(&B43C+n\8)\2^(n MOD 8)) MOD 2
                          a = PEEK(&B34C+n)
                          b = PEEK(&B39C+n)
                          c = PEEK(&B3EC+n)

MODE n                    n = PEEK(&B1C8)

ORIGIN x,y,a,b,c,d        x = UNT(PEEK(&B328)+256*PEEK(&B329))
                          y = UNT(PEEK(&B32A)+256*PEEK(&B32B))
                          a = (PEEK(&B330)+256*PEEK(&B331))
                              *2^(2-PEEK(&B1C8))
                          b = (PEEK(&B332)+256*PEEK(&B333))
                              *2^(2-PEEK(&B1C8))
                          c = PEEK(&B334)*2
                          d = PEEK(&B336)*2

PAPER #n,i                wenn n=0:  j = PEEK(&B290)
                          wenn n>0:  j = PEEK(&B218+15*n)
                          weiter siehe Tabelle 2

PEN #n,i                  wenn n=0:  j = PEEK(&B28F)
                          wenn n>0:  j = PEEK(&B217+15*n)
                          weiter siehe Tabelle 2

PLOT x,y,i                x = XPOS und y = YPOS
                          j = PEEK(&B338)
                          weiter siehe Tabelle 2

PLOTR x,y,i               s. PLOT x,y,i

RAD                       s. DEG

SOUND a,b,c,d,e,f,g       a = PEEK(&ADB2)
                          b = PEEK(&ADB5)+256*PEEK(&ADB6)
                          c = PEEK(&ADB9)+256*PEEK(&ADBA)
                          d = PEEK(&ADB8)
                          e = PEEK(&ADB3)
                          f = PEEK(&ADB4)
                          g = PEEK(&ADB7)

SPEED INK a,b             a = PEEK(&B1D8)
                          b = PEEK(&B1D7)

SPEED KEY a,b             a = PEEK(&B4EA)
                          b = PEEK(&B4E9)

SPEED WRITE n             n = (PEEK(&B8D1)-6)/6

SYMBOL n,b(1),b(2),...    b(i) = PEEK(HIMEM+i+8*(n-PEEK(&B294)))

SYMBOL AFTER n            n = PEEK(&B294)

TAG #n                    wenn n=0:  PEEK(&B293)=0   -> TAGOFF
                                     PEEK(&B293)=255 -> TAG

                          wenn n>0:  PEEK(&B21B+15*n)=0   -> TAGOFF
                                     PEEK(&B21B+15*n)=255 -> TAG

TAGOFF #n                 s. TAG #n

WIDTH n                   n = PEEK(&AC24)

WINDOW #n,a,b,c,d         wenn n=0:  a = PEEK(&B289)+1
                                     b = PEEK(&B28B)+1
                                     c = PEEK(&B288)+1
                                     d = PEEK(&B28A)+1

                          wenn n>0:  a = PEEK(&B211+15*n)+1
                                     b = PEEK(&B213+15*n)+1
                                     c = PEEK(&B210+15*n)+1
                                     d = PEEK(&B212+15*n)+1

ZONE n                    n = PEEK(&AE79)

                    "Rückfahrkarte für Parameter"
```

Tabelle 1: Umwandlung von a(i) in b(i):

a(i)=	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
b(i)=	13	27	19	25	1	7	10	16	28	29	24	26	6	8	15	17

a(i)=	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
b(i)=	30	31	18	20	0	2	9	11	4	22	21	23	3	5	12	14

Tabelle 2: Umwandlung von j in i:

j=	0	3	12	15	48	51	60	63	192	195	204	207	240	243	252	255
MODE 0: i=	0	8	2	10	4	12	6	14	1	9	3	11	5	13	7	15
MODE 1: i=	0	-	-	2	-	-	-	-	-	-	-	-	1	-	-	3
MODE 2: i=	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1

(Wie diese Werte zustande kommen, findet man z.B. in "CPC International", 11/85, S. 34 f beschrieben)

Grafik Hardcopy für Anspruchsvolle

für 464-664-6128



Wohl jeder Druckerbesitzer hat irgendwann einmal den Wunsch gehegt, seine mühsam aufgebaute Bildschirmgrafik zu Papier zu bringen, und mit einer der vielen Hardcopyroutinen, die bereits veröffentlicht wurden, schien das auch kein Problem zu sein. Das Ergebnis entsprach dann jedoch meist nicht den Erwartungen, so daß diese Idee ersteinmal wieder in der Versenkung verschwand.
Aus dieser Erfahrung heraus veröffentlichen wir das nachfolgende Programm, das gegenüber den herkömmlichen Routinen einige wesentliche Verbesserungen aufzuweisen hat.
Zunächst einmal ist dies, unseres Wissens nach, die erste Hardcopy-Routine, die alle 8 Bits der Centronics-Schnittstelle nutzt.
Voraussetzung zum einwandfreien Betrieb ist, daß Sie in Ihrem Rechner eine der vielen Hardware-Erweiterungen für das achte Bit eingebaut haben (z.B. "Operation gelungen..." Heft 5/86). Selbstverständlich funktioniert es auch mit jeder anderen Erweiterung.

Weiterhin werden die Bildschirmfarben - anders als sonst üblich - auf dem Papier durch entsprechende Rasterung wiedergegeben, so daß Besitzer eines Grünmonitors eine fast 100 %-ige Kopie des Bildschirms erhalten. Da diese Rasterung in Mode 2 nicht nötig ist und in Mode 0 nicht realisiert werden kann, kann die Routine nur in Mode 1 aufgerufen werden.
Um eine wirklich verzerrungsfreie Grafik zu erhalten, wird ein besonderer Druckmodus verwandt, in dem der Drucker die Punkte horizontal in genau dem gleichen Abstand bringt wie vertikal.
Das Bild wird dabei um 90 Grad gedreht und im DIN-A4-Format ausgedruckt.
Die Routine erkennt den Rechnertyp und paßt sich selbständig an. Da auf die umständliche Pixeltest-Abfrage verzichtet wurde, arbeitet das Programm sehr schnell und kann jederzeit mit ESC abgebrochen werden.

Anpassung an verschiedene Drucker:
Das Programm wurde ursprünglich für den Star SG 10 geschrieben. Folgende Adressen ermöglichen eine Anpassung an andere Druckertypen :
&A09F
CODE: &1B &4D &0F (ESC "M" 15)
Funktion: Linken Rand auf Spalte 15
&A0A2
CODE: &1B &41 &08 (ESC "A" 8)
Funktion: Zeilenabstand 8/72 Inch
&A0A8
CODE: &1B &67 &05 (ESC "g" 5)
Funktion: Grafikausdruck mit 72 Pkt./Inch
&A0AB
CODE: &90 &01 (144 1)
Funktion: Es folgen 1*256+144 Grafikbytes
Der Wert &06 in &A09E und der Wert &07 in &A0A5 geben dem Rechner jeweils an, wieviel Steuercodes im folgenden an den Drucker geschickt werden müssen.

Anpassungsbeispiele:
EPSON FX 85:
51 POKE &A0A0,&6C:POKE &A0A9,&2A
SEIKOSHA SP-1000 CPC:
51 POKE &A0A0,&6C:POKE &A0A9,&2A:POKE &A0AC,&02
(T. Kochmann/TM)

```
10 REM hardcopy                                  [947]
20 SYMBOL AFTER 256:MEMORY &A000-1               [1742]
30 FOR i = &A000 TO &A0AC                        [963]
32 READ x$:IF x$="ab" THEN x$="a0"               [2151]
40 POKE i,VAL("&"+x$)                            [641]
50 NEXT                                          [350]
51 IF PEEK(&AC0E)<>&C9 THEN POKE &A01E,&C3       [9334]
:POKE &A01F,&B7:POKE &A02A,&C4:POKE &A02B,
&B7:PRINT"664/6128 version" ELSE PRINT"464
version"
60 PRINT:PRINT"|hardcopy":PRINT                  [1887]
70 CALL &A000                                    [637]
80 DATA 01,09,ab,21,16,ab,c3,d1,bc               [1875]
90 DATA 0e,ab,c3,1a,ab,48,41,52,44,43,4f         [2108]
100 DATA 50,d9,00,00,00,00,fe,00,c0              [1898]
110 DATA 3a,c8,b1,fe,01,c0,21,9e,ab,cd,7f,       [1855]
ab
120 DATA 2a,c9,b1,11,4f,c0,19,7c,e6,07           [1629]
130 DATA f6,c0,67,06,50,c5,e5,e5,21,a5,ab        [1429]
140 DATA cd,7f,ab,e1,06,c8,c5,e5,4e              [1572]
150 DATA 06,04,cb,21,cb,1b,cb,3b,10,f8,06,       [2455]
04
160 DATA cb,21,cb,1a,cb,3a,10,f8,7a              [1071]
170 DATA ab,4f,cb,21,a3,47,cb,20,b1,f5,7a,       [1854]
b3
180 DATA b0,cd,88,ab,f1,cd,88,ab,e1,c1           [1133]
190 DATA cd,26,bc,10,cd,e1,c1,cd,23,bc           [1774]
200 DATA 10,ba,c9,46,23,7e,cd,88,ab              [2012]
210 DATA 10,f9,c9,f5,97,cd,1b,bb,fe,ef,28,       [2392]
```

```
09
220 DATA cd,2e,bd,38,f4,f1,c3,2b,bd          [1549]
230 DATA 31,f8,bf,c9,06,1b,4d,0f,1b,41,08    [1447]
240 DATA 07,0d,0a,1b,67,05,90,01             [1383]
```

Firmware-Erweiterung für 464-664-6128

Folgende kleine Routinen stellen Assemblerprogrammierern ein wertvolles Hilfsmittel bei der Entwicklung von Anwenderprogrammen zur Verfügung. Die Ein- und Ausgabe von Hexzahlen stellen ein nicht geringes Problem dar.

Eine genaue Beschreibung der neuen Calls finden Sie in der Mini-Referenz. Die Input-Routine verträgt nur Großbuchstaben und kann mit "ESC" wieder verlassen werden.
(TM)

```
10 DATA &3E,&26,&CD,&5A,&BB,&CD,&81,&BB,&C [4368]
D,&06,&BB,&FE,&FC,&C8,&FE,&2F,&DA,&58,&A0,
&FE, 3228
20 DATA &3A,&D2,&D9,&A0,&CD,&5A,&BB,&FE,&3 [5150]
A,&D2,&C1,&A0,&D6,&30,&07,&07,&07,&07,&E6,
&F0, 2762
30 DATA &F5,&CD,&81,&BB,&CD,&06,&BB,&FE,&F [4826]
C,&C8,&FE,&2F,&DA,&7C,&A0,&FE,&3A,&D2,&E1,
&A0, 3580
40 DATA &CD,&5A,&BB,&FE,&3A,&D2,&C6,&A0,&D [5379]
6,&30,&E6,&0F,&C1,&B0,&C9,&F5,&3E,&26,&CD,
&5A, 3079
50 DATA &BB,&F1,&F5,&0F,&0F,&0F,&0F,&E6,&0 [4308]
F,&FE,&0A,&D2,&CB,&A0,&C6,&30,&CD,&5A,&BB,
&F1, 2784
60 DATA &E6,&0F,&FE,&0A,&D2,&D3,&A0,&C6,&3 [4345]
0,&CD,&5A,&BB,&C9,&D6,&37,&C3,&72,&A0,&D6,
&37, 3026
70 DATA &C3,&96,&A0,&C6,&37,&CD,&5A,&BB,&C [4506]
3,&B3,&A0,&C6,&37,&CD,&5A,&BB,&C9,&FE,&41,
&D2, 3239
80 DATA &E9,&A0,&C3,&58,&A0,&FE,&41,&D2,&F [3427]
1,&A0,&C3,&7C,&A0,&FE,&47,&D2,&58,&A0,&C3,
&68, 3327
90 DATA &A0,&FE,&47,&D2,&7C,&A0,&C3,&8C,&A [1996]
0,&00, 1474
100 dat=0 : sz=0 : dz = 10                 [319]
110 FOR adr =-24496  TO-24327              [1142]
120   READ byte : dat=dat+1                [956]
130   sz=sz+byte                           [619]
140   POKE adr,byte                        [84]
150   IF dat < 20 AND adr < -24327 THEN 19 [1216]
0
160   READ chksum                          [1222]
170   IF chksum<>sz THEN PRINT "Fehler in  [3056]
Zeile :";dz
180   dz=dz + 10 : sz=0 : dat=0            [843]
190 NEXT adr                               [547]
200 END                                    [110]
```

Firmware-Beschreibung:

Adresse: &A050	Zahlenabfrage und Wartefunktion (Input Hex)
Aufgabe/Wirkung:	Ein "&"-Zeichen wird ausgegeben und eine 8-Bit-Zahl wird als Hex-Zahl abgefragt.
Einsprungsbedingungen:	keine
Übergabeparameter beim Aussprung:	A enthält Wert
Anmerkungen:	keine

Adresse: &A055	Zahlenabfrage und Wartefunktion (Input Hex).
Aufgabe/Wirkung:	Wie bei 42500, nur daß kein "&"-Zeichen ausgegeben wird.
Einsprungsbedingungen:	keine
Übergabeparameter beim Aussprung:	A enthält Wert.
Anmerkungen:	keine

Adresse: &A09B	Zahlenausgabe (Output Hex).
Aufgabe/Wirkung:	Gibt A als Hex-Zahl auf dem Bildschirm aus; sie wird durch ein "&"-Zeichen gekennzeichnet.
Einsprungsbedingungen:	A muß den auszugebenden Wert enthalten.
Übergabeparameter beim Aussprung:	keine
Anmerkung:	A wird zerstört.

Adresse: &A0A2	Zahlenausgabe (Output Hex).
Aufgabe/Wirkung:	Gibt A als Hex-Zahl am Bildschirm aus; sie wird nicht durch ein "&"-Zeichen gekennzeichnet.
Einsprungsbedingungen:	wie bei 42543.
Übergabeparameter beim Aussprung:	keine
Anmerkung:	A wird zerstört.

Befehle zur Einzelbitverarbeitung:
Es kann sehr nützlich sein, wenn man mit einem einzigen Befehl ein beliebiges Bit in einem beliebigen Register setzen oder zurücksetzen kann. Diese Möglichkeit beansprucht eine sehr große Zahl von OP-Codes und steht deshalb bei den meisten Mikroprozessoren nicht zur Verfügung. Der Z-80 jedoch ist mit ausgiebigen Möglichkeiten zur Bitmanipulation ausgerüstet.
Hauptsächlich gibt es drei Möglichkeiten:
1. Bit testen (BIT ...)
Wird ein Befehl zum Testen eines Bits ausgeführt, so enthält nach Ausführung des Befehls das Zero-Flag das Komplement des getesteten Bits.
2. Bit setzen (SET ...)
Nach Ausführung des Befehls ist das angesprochene Bit gesetzt.
3. Bit löschen (RES ...)
Nach Ausführung des Befehls ist das angesprochene Bit gelöscht.
Grundsätzlich gibt es hierbei zwei Arten von Operanden:
1. Welches Bit ist gemeint?
Hier muß eine Zahl zwischen 0 und 7 angegeben werden.
2. Wo ist dieses Bit?
Hierbei gibt es mehrere Möglichkeiten, die auch zur Ausführung unterschiedliche Zeiten benötigen. Sie sind jedoch bei allen drei Arten der Bitmanipulation gleich, so daß ich mich darauf beschränke, die Bit-Befehle (teste Bit) zu erklären. Wird bei SET oder RES auf Speicherbereiche zugegriffen, so müssen 3 Taktzyklen zur Ausführungszeit addiert werden.
Bit b,r
b ist, wie unter 1. schon erklärt, eine Zahl zwischen 0 und 7, wobei die 7 das höchstwertige Bit darstellt.
r ist ein Einzelregister A, B, C, D, E, H oder L.
Die Ausführungszeit beträgt 8 Taktzyklen.
Bit b,[HL]
b ist, wie unter 1. schon erklärt, eine Zahl zwischen 0 und 7, wobei die 7 das höchstwertige Bit darstellt.
[HL] bedeutet, wie Ihnen sicher schon von anderen Befehlen her bekannt sein dürfte; die Zahl in HL wird als Adresse interpretiert und dieses Byte wird dann entsprechend behandelt.
Die Ausführungszeit beträgt 12 Taktzyklen.
Bit b,[IX+d]
b ist, wie unter 1. schon erklärt, eine Zahl zwischen 0 und 7, wobei die 7 das höchstwertige Bit darstellt.
IX wird wie in [HL] als Adresse interpretiert. Dazu wird allerdings der 8-Bit-Wert d addiert. Die dadurch entstehende Adresse ist dann der Operand.
Die Ausführungszeit dauert, wie bei allen Index-Befehlen, relativ lange - 20 Taktzyklen.
Bit b,[IY+d]
Dies ist im Prinzip der gleiche Befehl, allerdings wird statt des IX-Registers das IY-Register zur Indizierung benutzt.
Die Ausführungszeit ist ebenfalls 20 Taktzyklen.
Die CPU-Steuerbefehle:
CCF
Mit Hilfe dieses Befehls läßt sich das Carry-, auch Übertragsflag genannt, komplementieren. Die Ausführungszeit dauert 4 Taktzyklen.
SCF
Dieser Befehl setzt das Übertragsflag. Die Ausführung dauert ebenfalls nur 4 Taktzyklen.
NOP
Ein Befehl, den die meisten Prozessoren kennen. Er bedeutet "No Operation", und wie der Name schon andeutet, hat dieser Befehl keine Funktion. Er wird meist bei Änderungen angewandt, um freiwerdende Bytes keiner Funktion auszusetzen, oder bei zeitkritischen Anwendungen entsprechend zu verweilen. Die Ausführungszeit dieses nichts machenden Befehls dauert 4 Taktzyklen (eine Interpretationszeit ist immer notwendig).
HALT
Auch dieser Befehl läßt sich einfach erklären. Die CPU wird zum Stillstand gebracht - der Name deutet es ja schon an. Sie wird erst wieder aktiv, wenn ein Interrupt (INT oder NMI) oder der RESET-Eingang der CPU aktiv wird.
(Für Systemprogrammierer: Die Durchführung der Speicherrefreshs ist gewährleistet, da im Haltzustand NOP-Befehle ausgeführt werden.)
DI und EI
DI (disable Interrupt) verhindert die Ausführung eines maskierten Interrupts.
EI (enable Interrupt) gibt diese Sperre wieder frei.
Jeder der beiden Befehle benötigt 4 Taktzyklen zur Ausführung.
IM d
d ist eine Zahl zwischen 0 und 2. Dieser Befehl steuert die verschiedenen Interrupt-Betriebsarten. Die Ausführung dieses Befehls dauert 8 Taktzyklen.
0 - in dieser Betriebsart kann die unterbrechende Peripherie hardwaremäßig beliebige Anweisungen auf den Datenbus legen, die dann von der CPU ausgeführt werden.
1 - in dieser Betriebsart führt die CPU einen Restart-Befehl zu der Speicherstelle 56 dezimal aus.
2 - in dieser Betriebsart ist ein Speicher-indirekt arbeitender Betriebsroutinenaufruf möglich, wobei die angesprungene Bedienroutine in jedem beliebigen Bereich des 64 Kilobyte großen Arbeitsspeicher beginnen kann.
Sprungbefehle:
JP nn
Dies ist ein einfacher Jump. Es ist damit möglich, zu jeder Speicherstelle im gesamten 64K-Bereich des Z-80 zu springen. Jedoch muß die Adresse natürlich als 16-Bit-Wert abgelegt sein, und das kostet Speicherplatz.
Die Ausführung dauert 10 Taktzyklen.
JP b,nn
Dies ist ein bedingter Sprung. Je nach Stand der Flags wird entweder zu der angegebenen Adresse gesprungen oder, falls die Bedingung nicht zutrifft, der Sprungbefehl wird ignoriert. Man könnte es mit folgender BASIC-Anweisung vergleichen:
1200 IF FLAG=0 THEN 1000
b ist eine Bedingung:
NZ - nicht Null
Z - Null
NC - kein Übertrag
C - Übertrag
PO - gerade Parität
PE - ungerade Parität
P - Vorzeichen positiv
M - Vorzeichen negativ
nn ist eine 16-Bit-Adresse.
Die Ausführungszeit dauert 10 Taktzyklen.
JR n
Dies ist ein relativer unbedingter Sprung. Die Sprungweiten wird zum momentanen Stand des Befehlszählers addiert (im Zweierkomplement) und ermöglicht so einen Sprung im Bereich zwischen -126 und +129 Bytes. Er dauert in seiner Ausführung 12 Taktzyklen, also trotz kürzeren OP-Codes längere Ausführungszeit. Von Vorteil ist aber, daß ein Programm, das nur den JR als Sprungbefehl benutzt, frei verschiebbar bleibt.
JR b,n
Dieser Befehl erlaubt einen relativen Sprung um n Bytes mit der Bedingung b. Die Sprungweite liegt im Bereich zwischen -126 und +129 Bytes.
b ist eine Bedingung:
NC - kein Übertrag
C - Übertrag
NZ - nicht Null
Z - Null
Die Ausführungszeit beträgt bei

zutreffender Bedingung, bei einer Sprungausführung also, 12 Taktzyklen. Sollte die Bedingung jedoch nicht zutreffen, nur 7 Taktzyklen.

JP [nn]

Der Inhalt des Doppelregisters nn wird als Adresse interpretiert. Dahin führt dann auch der Sprung.
nn kann sein:
HL - Ausführungszeit 4 Taktzyklen
IX - 8 Taktzyklen
IY - 8 Taktzyklen

DJNZ n

Dies ist ein Schleifenbefehl, der sehr häufig Verwendung findet. Bei jedem Erreichen des Befehls wird der Inhalt des Registers B um 1 erniedrigt. Nun wird geprüft, ob B den Wert 0 erreicht hat. Sollte dies nicht der Fall sein, so wird zu der angegebenen Adresse gesprungen. Andernfalls wird die Sprunganweisung ignoriert.
Beispiel:
LD B,80;
Schleifenanzahl festlegen
LOOP1 CALL AUSGAB;
Aufruf eines beliebigen Unterprogrammes
DJNZ LOOP1;
Ende der Schleife
Gewiß erkennen Sie Ähnlichkeiten mit einer BASIC "FOR-NEXT". Die Ausführungszeit beträgt 15 Taktzyklen, wenn die Schleife ausgeführt wird, und 8 Taktzyklen, falls sie nicht ausgeführt wird.

CALL nn

Dies ist ein Unterprogrammaufruf, ähnlich einem "GOSUB" in BASIC. Bei Erreichen dieses Befehls wird zuerst der aktuelle Befehlszählerstand (Inhalt des PC-Registers) auf den Stapel gerettet. Dies geschieht in zwei Phasen (eine Adresse sind 16 Bits). Anschließend wird die Adresse nn angesprungen.
Die bei nn beginnende Subroutine muß durch "RET" abgeschlossen sein. Dieser Befehl restauriert den Befehlszähler aus dem Stack wieder auf die nächste Adresse des Hauptprogrammes.
Die Ausführungszeit dauert 17 Taktzyklen.

CALL b,nn

Ein bedingter Unterprogrammaufruf. Trifft die Bedingung b zu, so wird der Sprung ausgeführt und dauert 17 Taktzyklen, andernfalls wird der Unterprogrammaufruf ignoriert und dauert nur 10 Taktzyklen.
b kann sein:
NZ - nicht Null
Z - Null
NC - kein Übertrag
C - Übertrag
PO - gerade Parität
PE - ungerade Parität
P - Vorzeichen positiv
M - Vorzeichen negativ

RST n

Dies ist ebenfalls ein Unterprogrammaufruf. Er kann allerdings nur dazu benutzt werden, bestimmte Adressen anzuspringen. Er ist in seiner Ausführungszeit mit 11 Taktzyklen wesentlich schneller als ein "CALL". Dies ist bedingt durch seinen nur 1 Byte großen OP-Code.
n kann sein:
00H
08H
10H
18H
20H
28H
30H
38H
n ist dann gleichzeitig die Adresse des Unterprogramms.

RET

Mit RET wird ein Unterprogramm abgeschlossen. Die beiden obersten Werte des Stacks werden in das PC-Register zurückkopiert. Die Ausführungszeit dauert 10 Taktzyklen.

RET b

Mit RET b ist ein bedingter Rücksprung aus einer Subroutine möglich. Ist b erfüllt, so erfolgt ein normaler Rücksprung. Andernfalls wird der Befehl ignoriert. Die Ausführung dauert mit Rücksprung 11, sonst 5 Taktzyklen.
b kann sein:
NZ - nicht Null
Z - Null
NC - kein Übertrag
C - Übertrag
PO - gerade Parität
PE - ungerade Parität
P - Vorzeichen positiv
M - Vorzeichen negativ

RETI

Dieser Befehl bedeutet "Return from Interrupt". Ein Interrupt ist so etwas ähnliches wie ein Unterprogrammaufruf. Er funktioniert allerdings etwas komplizierter. Zwei Aufgaben werden durch RETI erfüllt:

1. Der Programmzähler wird analog zu RET restauriert.
2. Dem Peripheriebaustein, der den INT anmeldete, wird das Ende seiner zugehörigen Routine mitgeteilt. Dieser Baustein gibt daraufhin die von ihm blockierte DAISY-CHAIN wieder frei und ermöglicht damit die Abarbeitung niederwertiger INTs. Darüber hinaus wird das IFF2- in das IFF1-Flip-Flop kopiert.

Beachten Sie jedoch, daß durch die RETI-Anweisung der maskierbare Interrupt nicht freigegeben wird. Daher sollte grundsätzlich vor jedem RETI-Aufruf ein EI-Befehl stehen, der die Annahme später folgender Interruptanforderungen ermöglicht. Die Ausführung dauert übrigens 14 Taktzyklen.

RETN
RETN wird zur Rückkehr aus einer Unterroutine verwendet, die durch einen nicht maskierbaren Interrupt (NMI) aufgerufen wurde. RETN ist in seiner Wirkung vollkommen identisch zum Befehl RET, zusätzlich wird jedoch das IFF2-Flip-Flop in das IFF1-Flip-Flop kopiert, so daß die Bearbeitung maskierbarer Interruptanforderungen unmittelbar nach Ausführung des RETN-Befehls freigegeben ist, falls sie bereits vor Auftreten der NMI-Anforderung freigegeben war. Die Ausführung dauert ebenfalls 14 Taktzyklen.

Ein- und Ausgabebefehle:
IN A,[n]
Der Operand n belegt die untere Hälfte des Adressbusses (A0 bis A7) und wählt eine I/O-Schnittstelle mit der Adresse n aus. Der Inhalt vom Akkumulator belegt gleichzeitig die obere Adresshälfte (A8 bis A15). Anschließend wird ein Byte der gewählten I/O-Schnittstelle auf den Datenbus übernommen und in den Akkumulator transferiert. Die Ausführung dieses Befehls dauert 11 Taktzyklen.
IN r,[C]
Der Inhalt des Registers C erscheint auf der unteren Adresshälfte (A0 bis A7), Register B auf der oberen (A8 bis A15). Die durch den Inhalt des C-Registers adressierte Schnittstelle liefert das in ihr enthaltene Byte über den Datenbus in das gewünschte Register r. Diese Befehlsausführung dauert 12 Taktzyklen.
INI
Die Schnittstelle, deren Adresse in C enthalten ist, wird abgefragt. Ihr Inhalt gelangt über den Datenbus in jene Speicherstelle, deren Adresse in HL-Register enthalten ist. Anschließend wird der Inhalt von HL inkrementiert. Das Register B wird dekrementiert.
Also:
(HL) ← (C)
B ← B-1
HL ← HL+1
Die Ausführung dieses Befehls dauert 16 Taktzyklen.
INIR
Mit diesem Befehl kann eine fortlaufende Dateneingabe über eine Schnittstelle, deren Adresse in C steht, durchgeführt werden. Die Anzahl der zu übernehmenden Bytes wird mit B festgelegt. Die Startadresse des Datenblocks ist durch den Inhalt von HL gegeben.
Die Ausführungszeit dauert, wenn B =0, 16 Taktzyklen, wenn B <> 0, 21 Taktzyklen.
Also:
INIR ist eine fortlaufende Aktion des INI-Befehls. Die gesamten Operationen des INI-Befehls werden so lange durchgeführt, bis der Inhalt des B-Registers gleich Null ist. Ist B <> 0, wird der Programm-Counter um zwei erniedrigt und der INI-Befehl erneut durchgeführt. Interruptanforderungen werden nach jedem Datentransfer angenommen. Außerdem werden zwei Refresh-Zyklen ausgesandt.
IND
IND entspricht in seiner Funktion dem INI-Befehl, mit dem Unterschied, daß nach Dateneingabe dekrementiert anstatt inkrementiert wird.
Also:
(HL) ← (C)
B ← B-1
HL ← HL-1
Die Ausführungszeit dauert 16 Taktzyklen.
INDR
Der Befehl INDR entspricht dem IND-Befehl. Allerdings wird er so oft wiederholt, solange Register B ungleich Null ist. Die Ausführungszeit bei B=0 beträgt 16 Taktzyklen, bei B<> 0 21 Taktzyklen.
OUT [n],A
Der Operand n belegt die untere Hälfte des Adressbusses (A0 bis A7) und wählt eine I/O-Schnittstelle mit der Adresse n aus. Der Inhalt vom Akkumulator belegt die obere Adresshälfte (A8 bis A15). Anschließend wird ein Byte aus dem Akku auf die gewählte I/O-Schnittstelle auf den Datenbus übergeben. Die Ausführung dieses Befehls dauert 11 Taktzyklen.
OUT [C],r
Der Inhalt des Registers C erscheint auf der unteren Adresshälfte (A0 bis A7), Register B auf der oberen (A8 bis A15). Auf die durch den Inhalt des C-Registers adressierte Schnittstelle wird der Wert des Registers r ausgegeben. Diese Befehlsausführung dauert 12 Taktzyklen.
OUTI
Der Inhalt des Doppelregisters HL wählt eine Speicheradresse, deren Inhalt wird über die Schnittstelle, deren Adresse in C enthalten ist, ausgegeben. Nach Übernahme des Inhalts der Zelle in die CPU, wird der Bytezähler B dekrementiert. Anschließend gelangt der Inhalt von C auf die untere, der von B auf die obere Adressbushälfte. Der zwischengespeicherte Inhalt der Speicherzelle wird nun auf den Datenbus gebracht und über die ausgewählte Schnittstelle ausgegeben. Danach wird das Registerpaar HL inkrementiert.
Also:
(C) ← (HL)
B ← B-1
HL ← HL+1
Die Ausführung dieses Befehls dauert 16 Taktzyklen.
OTIR
Mit diesem Befehl kann eine fortlaufende Dateneingabe über eine Schnittstelle, deren Adresse in C steht, durchgeführt werden. Die Anzahl der zu übernehmenden Bytes wird mit B festgelegt. Die Startadresse des Datenblocks ist durch den Inhalt von HL gegeben.
Die Ausführungszeit dauert, wenn B=0, 16 Taktzyklen, wenn B<> 0, 21 Taktzyklen.
Also:
OTIR ist eine fortlaufende Aktion des OUTI-Befehls. Die gesamten Operationen des OUTI-Befehls werden so lange durchgeführt, bis der Inhalt des B-Registers gleich Null ist. Ist B <> 0, wird der Programm-Counter um zwei erniedrigt und der OUTI-Befehl erneut durchgeführt. Interruptanforderungen werden nach jedem Datentransfer angenommen. Außerdem werden zwei Refresh-Zyklen ausgesandt.
OUTD
OUTD entspricht in seiner Funktion dem OUTI-Befehl, mit dem Unterschied, daß nach Datenausgabe dekrementiert anstatt inkrementiert wird.
Also:
(C) ← (HL)
B ← B-1
HL ← HL-1
Die Ausführungszeit dauert 16 Taktzyklen.
OTDR
Der Befehl OTDR entspricht dem OUTD-Befehl. Allerdings wird er so oft wiederholt, solange Register B ungleich Null ist. Die Ausführungszeit bei B=0 beträgt 16 Taktzyklen, bei B<> 0 21 Taktzyklen.

Ich hoffe, daß Ihnen unser Assemblerkurs gefallen hat und daß Sie aus der Zusammenfassung aller regulären Z-80-Befehle Nutzen ziehen werden. Ich möchte mich noch für Ihre Aufmerksamkeit bedanken und wünsche fröhliches Assemblerprogrammieren. (HF)

COPYPIC

LOGO-Grafik schwarz auf weiß

Leider bietet die Programmiersprache LOGO keine Möglichkeit, ein erstelltes "Bild" auf dem Drucker auszugeben. Die durch gleichzeitiges Drücken der EXTRA- und PTR-Taste ausgelöste Hardcopy-Funktion bringt naturgemäß den gesamten aktuellen Bildschirminhalt auf das Papier, und dies in einem vergleichsweise kleinen Format.
Mit COPYPIC kann man nun LOGO-Grafiken in einem größeren Format auf den Drucker ausgeben. Da die Anzahl der Kopfzeilen und der Rand unter Programmkontrolle einstellbar sind, lassen sich LOGO-Grafiken auch in Loco-Script-Texte "einbinden".

LOGO und seine Bilder

Wie aus dem "mageren" LOGO-Teil des Joyce-Benutzerhandbuches zu entnehmen ist, wird das gerade auf dem Schirm gezeigte Bild mit dem Befehl **savepic "name** in die Datei **name** auf Diskette gespeichert. Wer schon einmal mit DIR [SIZE] unter CP/M seine LOGO-Dateien begutachtet hat, wird bemerkt haben, daß jedes auch noch so kleine Bild volle 23 kByte Speicherplatz verbraucht. (Anmerkung für "Neulinge": bei dem DIR-Befehl muß statt den eckigen Klammern das große Ä für Klammer auf bzw. das große Ü für Klammer zu verwendet werden.) Wer nachrechnet, hat schnell die Lösung: Die Auflösung des Joyce-Monitors beträgt 720 x 256 Punkte (Spalten x Zeilen). Das macht 184320 Punkte, oder Pixel, wie die Eingeweihten einen Punkt auf dem Monitor nennen. Ein Pixel entspricht einem Bit, ein Byte enthält 8 Bit. Also 184320 geteilt durch 8 sind 23040 Bytes. Das entspricht den angezeigten 23 kByte.
LOGO tut aber noch mehr. Als Datei-Kopf werden 38 Byte abgespeichert. In den ersten sieben Bytes dieses Datei-Kopfes steht im ASCII-Code das Wort "PICTURE". Hinter dem Datei-Kopf sind dann die 256 Zeilen à 90 Byte abgespeichert. 720/8 = 90 Byte entsprechend der horizontalen Auflösung, 256 Zeilen entsprechend der vertikalen Auflösung.

Der Drucker und seine Grafik

Der Drucker von Joyce hat nicht nur einen immensen Zeichensatz und diverse Schriftarten, sondern er kann auch Grafik auf's Papier bringen. Das geht mit einfacher Auflösung (480 Pixel/Zeile) oder mit doppelter Auflösung (960 Pixel/Zeile). Da der Druckkopf des Druckers acht Nadeln besitzt, die untereinander (vertikal) angeordnet sind, druckt er immer acht Grafikzeilen auf einmal.

Der springende Punkt

In diesen, scheinbar nebensächlichen Anmerkungen liegt nun der Hund bzw. das Pixel begraben. Während LOGO in seiner Bild-Datei jeden Bildpunkt fein säuberlich hintereinander ablegt, braucht der Drucker acht Pixel, die jeweils 720 Bit bzw. 90 Byte auseinanderliegen (siehe auch Bild 1 und 2). Damit liegt die Aufgabenstellung für das Programm fest.

Bit für Bit für Bit...

Zunächst müssen acht Grafikzeilen à 90 Byte mit den 720 Bildpunkten von der Diskette gelesen werden. Dann werden jeweils aus den acht untereinanderliegenden Bytes, die eine 8x8-Bit-Matrix bilden, die Werte der Spalten berechnet und diese an den Drucker ausgegeben.
Für ein vollständiges LOGO-Bild muß der Drucker also 256 / 8 = 32 Zeilen drucken. Da nun erstens solche Bit-Rechnerei nicht gerade die Stärke von Basic ist, und zweitens pro Druckzeile 5760 Bildpunkte umgewandelt werden müssen, dauert die Erzeugung einer Druckzeile ca. eine Minute.

Das gewisse Extra

Der Gemütsmensch wird sich angesichts dieses Zeitaufwandes zur Kaffeepause zurückziehen und dem Computer die Arbeit überlassen. Wer es eiliger hat, dem bietet das Programm zwei Möglichkeiten immenser Beschleunigung:

1. Ein LOGO Bild nimmt in den seltensten Fällen den ganzen Bildschirm in Anspruch. Es ist also möglich, die Anzahl der Druckzeilen einzustellen.
2. Wenn die Daten für den Drucker schon umgewandelt werden müssen, dann kann man sie auch gleich in passender Form auf die Diskette schreiben. Eine solche Datei ist "ruckzuck" wieder ausgegeben.

Copy mit Komfort

Bei der Entwicklung des Programmes wurde darauf geachtet, daß Eingaben auf das unbedingt notwendige Maß reduziert wurden. Nach Eingabe und Start mit RUN, (Eingabe vor Starten des Programmes unbedingt speichern!!) zeigt sich ein viergeteilter Bildschirm.
Im ersten Teil finden sich außer dem Namen und der Versionsnummer die Werte einiger Parameter, die weiter unten besprochen werden. Im zweiten Teil befinden sich neun Menuepunkte, die zeigen, wie sich das Programm die weitere Zusammenarbeit mit dem Benutzer vorstellt. Der dritte und größte Teil des Bildschirmes ist dem Inhaltsverzeichnis vorbehalten. Beim ersten Start wird dieses in aller Regel die lapidare Meldung "keine Daten" beinhalten. Der untere Teil gibt Fehlermeldungen aus und informiert über Fortschritte beim Ausdruck eines Bildes.

Die Menuepunkte

Im zweiten Teil, in dem sich die Menuepunkte befinden, sehen Sie einen Menuepunkt invers dargestellt. Dort befindet sich der Menuecursor. Diesen können Sie mit den Cursor-links- bzw. -rechts-Tasten bewegen.
Sicher haben Sie auf irgendeiner Diskette schon ein LOGO-Bild abgespeichert. Diese LOGO-Bilder nennen wir ab jetzt PIC-Dateien, da sie den Dateityp .PIC besitzen.

Jetzt gehts los...

- Programmdiskette entnehmen.
- Datendiskette einlegen (die mit den PIC-Dateien).
- Mit dem Menuecursor auf den ersten Menuepunkt "diskette" fahren und die RETURN- oder ENTER-Taste drükken (alle Menuefunktionen werden so ausgelöst).

Nach kurzer Zeit erscheint auf dem Bildschirm im dritten Teil das Inhaltsverzeichnis der Diskette. Allerdings werden zu Ihrer Entlastung nur solche Dateien angezeigt, die das Dateiattribut PIC oder PRT besitzen. Die PIC-Dateien sind in der ersten, die PRT-Dateien in der zweiten Spalte aufgelistet. Die oberste PIC-Datei ist invers dargestellt. Dies ist der Dateicursor, mit dem die zu druckende Datei ausgewählt wird. Er kann mit den Cursor-hoch- bzw. -runter-Tasten bewegt werden, jedoch nur innerhalb der Gruppe mit den PIC-Dateien. Um in die Gruppe mit den PRT-Dateien zu gelangen, setzen Sie einfach den Menuecursor auf "PRT" und

Bild 1



So speichert LOGO ein Grafikbild auf der Diskette.

drücken RETURN. Da Sie aber vorläufig noch keine PRT-Datei im Inhaltsverzeichnis haben, ist diese Funktion noch sinnlos und der Dateicursor bleibt in der ersten Spalte. Mit dem Dateicursor können Sie also immer das Bild auswählen, das Sie ausdrucken lassen wollen.

PIC oder PRT

Die PIC-Dateien enthalten die Original-LOGO-Bilder, wie sie mit dem savpic-Befehl gespeichert wurden. PRT-Dateien hingegen sind Dateien, welche das Programm während des Ausdruckens einer PIC-Datei selbständig erzeugt. PRT-Dateien enthalten also die "schnellen" Bilder, von denen weiter oben schon die Rede war.

Darfs auch ein bißchen mehr sein?

Mit dem Menuepunkt "zeilen" können Sie einstellen, wieviele Druckzeilen der PIC-Datei ausgedruckt werden sollen. Maximal sind dies 32 Zeilen.
- Menuecursor auf "zeilen" stellen.
- RETURN drücken.
- Anstelle des inversen Gleichheitszeichens erscheint ein Fragezeichen und die gewünschte Zeilenzahl kann eingegeben werden.
- RETURN schließt die Eingabe ab.

Auf gleiche Art und Weise funktionieren die Menuepunkte "kopf" und "rand". Mit "kopf" läßt sich die Anzahl der Zeilenvorschübe vor Ausdruck des Bildes eingeben. Mit dem Menuepunkt "rand" wird die Breite des linken Randes eingestellt. Probieren hilft hier, das richtige Maß zu finden. Diese Parameter können ebenso für den Ausdruck einer PRT-Datei bestimmt werden.

Mit oder ohne?

Mit Hilfe des Menuepunktes "±prt" können Sie entscheiden, ob eine PRT-Datei erzeugt werden soll oder nicht. Normalerweise ist die Erzeugung einer PRT-Datei immer empfehlenswert. Sie müssen nur darauf achten, daß auf Ihrer Datendiskette genügend Speicherplatz vorhanden ist. Ein DISK-FULL-Fehler wird nicht abgefangen und bedeutet den unsanften Abbruch des Programmes. Eine PRT-Datei verbraucht nicht generell 23 kByte Speicherplatz; der Platzbedarf richtet sich nach der Anzahl der Zeilen, die gedruckt werden sollen und liegt zwischen einem kByte und 23 kByte. Die Laufzeit des Programmes verlängert sich durch die Generierung der PRT-Datei nur unmerklich. Die PRT-Datei wird mit demselben Namen versehen, den auch die PIC-Datei hat; es wird nur automatisch der Dateityp in .PRT geändert.

Let's go

Wenn Sie alle Parameter entsprechend Ihren Wünschen eingegeben haben, der Dateicursor an der richtigen Stelle steht und das Kaffeewasser kocht, bleibt nur noch eins zu tun:
- Menuecursor auf "START" stellen und RETURN-Taste drücken - und Kaffee trinken...

Zum Verlassen von COPYPIC in Richtung C/PM dient der Menuepunkt "EXIT"; abbrechen können Sie das Programm jederzeit mit der STOP-Taste.

COPYPIC intern

Im Listing sind ab Zeile 10000 die Bildschirm- und Druckerkommandos definiert. Da dieser Teil als Modul hinzugeladen wurde, werden von diesen Kommandos nicht alle benötigt. Wenn Sie diese zuerst eingeben und als Text-Datei mit **SAVE "BDCODES.TXT",A** abspeichern, so kommen Sie gleichzeitig zu einer recht umfassenden Kommandobibliothek, die bei der Entwicklung eigener Programme nützlich ist. In Ihre eigenen Programme können diese Kommandos mit **MERGE "BDCODES.TXT"** leicht eingebunden werden und mit **GOSUB 10000** initialisiert werden. Natürlich dürfen in Ihrem Programm die Zeilennummern von 10000 bis 11350 nicht benutzt werden.

(Thorsten Schlote/MC)

```
Bild 2

Z 1       |7|     |_|          8 |_|
Z 2    B  |6|  B  |_|          Y |_|
Z 3    Y  |5|  Y  |_|          T |_|
Z 4    T  |4|  T  |_|          E |_|
Z 5    E  |3|  E  |_|    ...     |_|
Z 6       |2|     |_|          7 |_|
Z 7    1  |1|  2  |_|          2 |_|
Z 8       |0|     |_|          0 |_|


Z 9       |_|
Z 10      |_|     ...
In dieser Reihenfolge bekommt der Drucker Daten.
```

```
1000  '________________________________________
1010  '
1020  '  COPYPIC (LOGO GRAFIK-AUSDRUCK)
1030  '  VERSION 1.04  /  APRIL 86
1040  '  THORSTEN SCHLOTE
1060  '________________________________________
2000  '
2010  'INITIALISIEREN
2020  '
2030  CLEAR, &HF400, 512,3,1024
2040  '
2050  GOSUB 10000                          :'BILDSCHIRM-
KOMMANDOS
2060  GOSUB 11000                          :'DRUCKER-KOM
MANDOS
2070  '
2080  PRINT home$ cls$;                    :'BILDSCHIRM
LÖSCHEN
2090  LPRINT CHR$(24);res$                 :'DRUCKER ZUR
ÜCKSETZEN
2100  OPTION NOT TAB                       :'&H09 (TABUL
ATOR) UNTERDRÜCKEN
2110  WIDTH 90 : WIDTH LPRINT 255
2120  '
2130  DEFINT a-z
2140  '
2150  DIM f0     ( 7 )                     :'Masken BEI
BIT UMWANDLUNG
2160     FOR i=0 TO 7
2170        READ f0(i)
2180     NEXT
2190  '
2200  DIM wahl$   ( 8 )                    :'TEXTE FÜR M
ENÜ
2210  FOR i=0 TO 8
2220        READ wahl$(i)
2230     NEXT
2240  '
2250  DIM inhalt$(1,15)                    :'INHALTSVERZ
EICHNISS DER DISKETTE
2260  DIM i$      ( 1 )                    :'DATEITYPE
2270     i$(0)=".PIC"
2280     i$(1)=".PRT"
2290  DIM imax    ( 1 )                    :'MAXIMALE AN
ZAHL IM INHALTSVERZEICHNISS
2300  '
2310  DIM a$      ( 7 )                    :'8 SUB-ZEILE
N x 90 ZEICHEN
2320  '
2330  DIM b(7)                             :'Bitfeld für
Umwandlung
2340  '
2350  num$        ="0123456789"            :'NUMMERISCHE
WERTE
2360  gruppe      =0
2370  datei       =0                       :'DEFAULT FÜR
ERSTE ANZEIGE
2380  '
2390  zmax        =32                      :'DEFAULT FÜR
DRUCKZEILEN
2400  rand        =0                       :'DEFAULT FÜR
LINKEN RAND
2410  kopf        =0                       :'DEFAULT FÜR
ANZAHL DER KOPFZEILEN
2420  '
2430  prtmod      =-1                      :'DEFAULT FÜR
PRT DATEI
2440  prt$        ="+PRT dat"
2450  '                                     WEITERE VAR
IABLE WERDEN WÄHREND DER
2460  '                                     LAUFZEIT EI
```

```
NGEFÜHRT
2470   '
3000   '________________________________________________
________________________________
3010   '
3020   ' HAUPTPROGRAMM
3030   '
3040   GOSUB 8010
3050   GOSUB 7010
3060   END
4000   '________________________________________________
________________________________
4010   '
4020   ' PRT ODER PIC DATEI DRUCKEN
4030   '
4040   ON gruppe+1 GOSUB 4510,4100
4050   RETURN
4060   '________________________________________________
________________________________
4100   '
4110   ' PRT DATEI DRUCKEN
4120   '
4140   OPEN "R" , #1 , inhalt$(gruppe,datei) , 240
4150   FIELD #1 , 240 AS gwert$
4160   GET #1
4170   '
4180   test$=LEFT$(gwert$,7) : x$="PICTURE"
4190      IF NOT (test$=x$) THEN ex=1 : GOSUB 9100
 : RESET : RUN
4200   x=ASC(MID$(gwert$,8,1))
4210      IF zmax > x THEN zmax=x
4220   '
4230   FOR i=1 TO kopf : LPRINT lf$; : NEXT
4240   '
4250   LPRINT FNlr$(rand);
4260   LPRINT FNn72$(8);
4270   '
4280   ' ZEILEN LESEN UND DRUCKEN
4290   '
4300      FOR zeile =1 TO zmax
4310         PRINT FNat$(28,5);"Z=";zeile
4320         LPRINT FNgd$(720);
4330         '
4340         ' 720 BYTES LESEN
4350         '
4360         FOR i=0 TO 2
4370            GET #1
4380            LPRINT gwert$;
4420         NEXT
4430      LPRINT
4440   NEXT zeile
4450   CLOSE #1
4460   RETURN
4500   '________________________________________________
________________________________
4510   '
4520   ' DRUCKEN UND PRT DATEI ERZEUGEN
4530   '
4540      IF kd THEN RETURN
4550      '
4560      IF prtmod THEN GOSUB 8800     :'PRT DATEI
 ÖFFNEN
4570      '
4580      ' BILD-DATEI ZUM LESEN ÖFFNEN
4590      '
4610      OPEN "R" , #1 , inhalt$(gruppe,datei) ,
1
4620      FIELD #1 , 1 AS byte$
4630      GOSUB 8500
4640         FOR i=1 TO kopf : LPRINT lf$; : NEXT
4650      LPRINT FNlr$(rand);
4660      LPRINT FNn72$(8);
4670      '
4680      ' ZEILEN LESEN UND DRUCKEN
4690      '
4700         FOR zeile=1 TO zmax
4710            PRINT FNat$(28,5);"Z=";zeile
4720            '
4730            LPRINT FNgd$(720);
4740            '
4750            ' 8 SUB-ZEILEN LESEN
4760            FOR subzeile = 0 TO 7 : a$(subzeil
e)=""
4765               FOR i=1 TO 90
4770                  GET #1 : a$(subzeile) = a$(s
ubzeile)+byte$
4775               NEXT
4780            NEXT
4790            ' EINGELESENE SPALTE UMWANDELN UND
 DRUCKEN
4800
4810                     FOR spalte=1 TO 90
4820                        PRINT FNat$(28,23);"S=";spal
te
4830                        '
4840                        ' UMWANDLUNG IN BITFELD
4850                        '
4855                        FOR i=0 TO 7 : b(i)=ASC(MID$
(a$(i),spalte,1)) : NEXT
4860                           FOR i=0 TO 7 : e=0
4870                              FOR j=0 TO 7
4880                                 IF b(j) AND f0(i) T
HEN e=e+f0(j)
4890                              NEXT
4940                              '
4950                              ' ZEICHEN DRUCKEN
4960                              '
4970                              LPRINT CHR$(e);
4980                                 IF NOT prtmod THEN
5010
4990                                    p$=p$+CHR$(e) :
pp=pp+1
5000                                       IF pp=241 THE
N GOSUB 5090
5010                           NEXT
5020                           '
5030                        NEXT
5040                     LPRINT
5050                  NEXT zeile
5060               CLOSE #1
5070                  IF prtmod THEN GOSUB 9000 :'PRT DATEI
 SCHLIESSEN
5080   RETURN
5090      '
5100      ' EINTRAG IN PRT DATEI
5110      '
5130         LSET pwert$=p$
5140         PUT #2
5150         p$="" : pp=1
5170      RETURN
5500   '________________________________________________
________________________________
5510   '
5520   ' VORBESETZUNG
5530   '
5540   DATA 128,64,32,16,8,4,2,1
        :'FELD  f0
5550   DATA "diskette","  PIC   "," PRT  "," zei
len "   :'FELD  wahl$
5560   DATA "  rand   ","  kopf   "," +-prt " ," STA
RT "
5570   DATA "  EXIT   "
6000   '________________________________________________
________________________________
6010   '
6020   ' WAHLEN
6030   '
6040   x=0
6050         IF wahl<7 THEN wahl=wahl+1 ELSE wahl=
gruppe+1
6060      WHILE x<>13
6070      PRINT FNat$(6,0);
6080         FOR i=0 TO 8
6090               IF i=wahl THEN PRINT invon$;
6100            PRINT FNat$(6,i*10); wahl$(i); inv
off$
6110         NEXT
6120      GOSUB 6210
6130      PRINT FNat$(3,89);
6140      x=ASC(INPUT$(1))
6150         IF x=6 THEN wahl=wahl+1 : IF wahl=9 T
HEN wahl=0 : GOTO 6190
6160            IF x=1 THEN wahl=wahl-1 : IF wahl=
-1 THEN wahl=8: GOTO 6190
6170                  IF x=31 THEN GOSUB 6310
6180                     IF x=30 THEN GOSUB 6380
6190      WEND
6200   RETURN
6210      '
6220      ' AKKTUELLE DATEI INVERS SETZEN
6230      '
6240      PRINT FNat$(9+datei,30*gruppe);invon$;in
halt$(gruppe,datei);invoff$;
6250      RETURN
6260      '
6270      ' AKTUELLE DATEI NORMAL SETZEN
6280      '
6290      PRINT FNat$(9+datei,30*gruppe);invoff$;i
nhalt$(gruppe,datei);
```

```
6300       RETURN
6310       '
6320       ' CURSOR UP   EINE DATEI HöHER
6330       '
6340       GOSUB 6260
6350       datei=datei-1
6360          IF datei<0 THEN datei=0
6370       RETURN
6380       '
6390       ' CURSOR DOWN EINE DATEI TIEFER
6400       '
6410       GOSUB 6260
6420       datei=datei+1
6430          IF datei>imax(gruppe) THEN datei=imax
(gruppe)
6440       RETURN
7000    '______________________________________________________
_____________________________________
7010    '
7020    ' HAUPTMENü
7030    '
7040       PRINT home$ cls$ st$
7050       PRINT
7060       PRINT "COPYPIC    Vers. 1.04";
7070       PRINT FNat$(3,30);invon$;"=";invoff$;zma
x;
7080       PRINT FNat$(3,40);invon$;"=";invoff$;ran
d;
7090       PRINT FNat$(3,50);invon$;"=";invoff$;kop
f;
7100       PRINT FNat$(3,60);prt$
7110       PRINT st$
7120       PRINT
7130       PRINT
7140       PRINT st$
7150       PRINT
7160          FOR i=0 TO 1
7170             FOR j=0 TO imax(i)
7180                PRINT FNat$(9+j,i*30);inhalt$(i
,j)
7190             NEXT
7200          NEXT
7210       PRINT FNat$(25,0);st$
7220       GOSUB 6000
7230          ON wahl+1 GOSUB 7250,7320,7380,7440,7
540,7640,7740,4010,7800
7240    GOTO 7010
7250       '
7260       ' DISKETTEN RESET
7270       '
7280    OPTION FILES "A"
7290       RESET
7300       GOSUB 8010
7310       RETURN
7320       '
7330       ' AUF PIC STELLEN
7340       '
7350       datei=0 : gruppe=0
7360          IF imax(0)=-1 THEN gruppe=1
7370       RETURN
7380       '
7390       ' AUF PRT STELLEN
7400       '
7410       datei=0 : gruppe=1
7420          IF imax(1)=-1 THEN gruppe=0
7430       RETURN
7440       '
7450       ' DRUCKZEILEN FüR GRAFIK EINSTELLEN
7460       '
7470       x=0
7480          WHILE x<1 OR x>32
7490             PRINT FNat$(3,30);"?   ";FNat$(3,3
2);
7500             GOSUB 8300
7510          WEND
7520       zmax=x
7530       RETURN
7540       '
7550       ' LINKEN RAND EINSTELLEN
7560       '
7570       x=-1
7580          WHILE x<0 OR x>19
7590             PRINT FNat$(3,40);"?   ";FNat$(3,4
2);
7600             GOSUB 8300
7610          WEND
7620       rand=x
7630       RETURN
7640       '
```

Grafik 1

```
7650       ' KOPFZEILEN EINSTELLEN
7660       '
7670       x=-1
7680          WHILE x<0 OR x>72
7690             PRINT FNat$(3,50);"?   ";FNat$(3,5
2);
7700             GOSUB 8300
7710          WEND
7720       kopf=x
7730       RETURN
7740       '
7750       ' OPTIONEN
7760       '
7770       prtmod=NOT prtmod
7780          IF prtmod THEN prt$="+PRT dat" ELSE p
rt$="-PRT dat"
7790       RETURN
7800    '
7810    ' EXIT
7820    '
7830    CLOSE
7840    PRINT home$ cls$;
7850    SYSTEM
7860    END
8000    '______________________________________________________
_____________________________________
8010    '
8020    ' INHALTSVERZEICHNISS ERSTELLEN
8030    '
8040    imax(0)=-1 : imax(1)=-1
8050    i=0
8060       WHILE i<2
8070          j=0
8080          inhalt$(i,j)=STRIP$(FIND$("*"+i$(i),j
+1))
8090             WHILE inhalt$(i,j)<>"" AND j<16
8100                j=j+1
8110                inhalt$(i,j)=STRIP$(FIND$("*"+i
$(i),j+1))
8120             WEND
8130          imax(i)=j-1
8140          i=i+1
8150       WEND
8160    kd=0
8170    gruppe=0
8180       IF imax(0)=-1 AND imax(1)>-1 THEN gruppe
=1
8190          IF imax(1)=-1 AND imax(0)>-1 THEN gru
ppe=0
8200             IF imax(0)=-1 AND imax(1)=-1 THEN
GOSUB 8220
8210    RETURN
8220       '
8230       ' BEIDE GRUPPEN KEINE DATEN
8240       '
8250       inhalt$(0,0)="Keine Daten"
8260       gruppe=0 : datei=0 : imax(0)=0
8270       kd=1
8280       RETURN
8300    '
8310    ' ZWEISTELLIGE ZAHL EINLESEN
8320    '
```

```
8330    x=2 : x$=" " : xe$=""
8340    DEF FNdel=(ASC(x$)=7 OR ASC(x$)=127)
8350       WHILE ASC(x$)<>13
8360          x$=INPUT$(1)
8370          IF x>0 AND INSTR(num$,x$)<>0 THEN PRI
NT x$; : x=x-1 : xe$=xe$+x$
8380          IF FNdel AND x=2 THEN x=2 : xe$="" :
PRINT " ";cleft$;
8390          IF FNdel AND x=1 THEN x=2 : xe$="" :
PRINT cleft$;" ";cleft$;
8400          IF FNdel AND x=0 THEN x=2 : xe$="" :
PRINT cleft$" "cleft$ cleft$" "cleft$;
8410       WEND
8420    x=VAL(xe$)
8430    PRINT FNat$(2,89);
8440    RETURN
8500    '
8510    ' AUSGEWÄHLTE DATEI PRÜFEN UND HEADER LESEN
8520    '
8530    gwert$=""
8540    FOR i=1 TO 38
8550       GET #1
8560       gwert$=gwert$+byte$
8570    NEXT
8580    test$=LEFT$(STRIP$(gwert$),7)
8590    IF NOT (test$="PICTURE") THEN ex=1 : GOSUB
9100 : RESET : RUN
8600    RETURN
8800    '
8810    ' PRT DATEI ERÖFFNEN
8820    '
8840    p$=""
8850    pp=1
8860    prtkopie$=LEFT$(inhalt$(gruppe,datei),8)+".
PRT"
8870    OPEN "R" , #2 , prtkopie$ , 240
8880    FIELD #2 , 240 AS pwert$
8890    x$="PICTURE"+CHR$(zmax)
8900    LSET pwert$=x$
8910    PUT #2
8930 RETURN
9000    '
9010    ' PRT DATEI SCHLIEßEN
9020    '
9040    LSET pwert$=p$
9050    PUT #2
9060    CLOSE #2
9070    GOSUB 8010
9080    RETURN
9100    '
9110    ' FEHLERBEHANDLUNG
9120    '
9130    PRINT FNat$(25,0);st$
9140    PRINT
9150    PRINT CHR$(7);
9160       IF ex<>1 THEN RETURN
9170    PRINT "Datei-Kopf ungültig in        :";inha
lt$(gruppe,datei)
9180    PRINT "Neu-Start mit  <RETURN>";
9190    x$=INPUT$(1) : RETURN
10000   REM ________________________________________
____________________________
10010   REM
10020   REM BILDSCHIRMKOMMANDOS
10030   REM ________________________________________
____________________________
10040                                    REM
10050   esc$       = CHR$(27):           REM ESCAPE
10060                                    REM

10070   home$      = esc$+"H":           REM CURSOR IN
 ZEILE 0, SPALTE 0
10080   eras$      = esc$+"J":           REM VON CURSO
R BIS ENDE SEITE LÖSCHEN
10090                                    REM
10100   cls$       = esc$+"E":           REM BILDSCHIR
M LÖSCHEN
10110   clz$       = esc$+"l":           REM ZEILE LÖS
CHEN
10120                                    REM
10130   cup$       = esc$+"A":           REM CURSOR EI
NE ZEILE HOCH
10140   cdown$     = esc$+"B":           REM CURSOR EI
NE ZEILE RUNTER
10150   cright$    = esc$+"C":           REM CURSOR EI
N ZEICHEN WEITER
10160   cleft$     = esc$+"D":           REM CURSOR EI
N ZEICHEN ZURÜCK
10170                                    REM
10180   getc$      = esc$+"j":           REM SPEICHERT
 CURSOR-POSITION
10190   prtc$      = esc$+"k":           REM SETZT CUR
SOR AUF GESPEICHERTE POSITION
10200                                    REM
10210   usoff$     = esc$+"u":           REM UNTERSTRE
ICHEN AUS
10220   uson$      = esc$+"r":           REM UNTERSTRE
ICHEN EIN
10230                                    REM
10240   invon$     = esc$+"p":           REM INVERS EI
N
10250   invoff$    = esc$+"q":           REM INVERS AU
S
10260                                    REM
10270   normal$    = esc$+"y":           REM 24x90 EIN
10280                                    REM
10290   st$=       STRING$(90,"_"):      REM STRICH ÜB
ER GANZEN BILDSCHIRM
10300                                    REM
10310                                    REM CURSOR AU
F ZEILE x, SPALTE y SETZEN
10320   DEF FNat$(x%,y%)              = esc$+"Y"+CHR$(
32+x)+CHR$(32+y)
10330                                    REM
10340                                    REM WINDOW MI
T x,y,h,b, DEFINIEREN
10350   DEF FNwindow$(x%,y%,h%,b%) = esc$+"X"+CHR$(
32+x)+CHR$(32+y)+CHR$(31+h)+CHR$(31+b)
10360   RETURN
11000   REM ________________________________________
____________________________
11010   REM
11020   REM DRUCKERKOMMANDOS
11030   REM ________________________________________
____________________________
11040                                              REM
11050   esc$=CHR$(27):                             REM
ESCAPE
11060                                              REM
11070                                              REM
SEITEN-LAYOUT
11080                                              REM
11090   za108zz$=esc$+"0":                         REM
8   ZEILEN/ZOLL
11100   za712zz$=esc$+"1":                         REM
7/12   ZEILEN/ZOLL
11110   za106zz$=esc$+"2":                         REM
6   ZEILEN/ZOLL
11120                                              REM
11130   DEF FNn216$(x)=esc$+"3"+CHR$(x):           REM
n/216  ZEILEN/ZOLL
11140   DEF FNn72$ (x)=esc$+"A"+CHR$(x):           REM
n/72   ZEILEN/ZOLL
11150                                              REM
11160   DEF FNlr$(x)=esc$+"l"+CHR$(x):             REM
LINKER  RAND
11170   DEF FNrr$(x)=esc$+"Q"+CHR$(x):             REM
RECHTER RAND
11180                                              REM
11190   cr$=CHR$(13):                              REM
WAGENRÜCKLAUF
11200   lf$=CHR$(10):                              REM
ZEILENVORSCHUB
11210   ff$=CHR$(12):                              REM
SEITENVORSCHUB
11220                                              REM
11230   einzbl$=esc$+"$"+esc$+"C"+"0"+CHR$(10):REM
EINZELBLATT DIN A4
11240   endlos$=esc$+"c"+esc$+"C0"+CHR$(11):       REM
ENDLOSPAP.  DIN A4
11250                                              REM
11260   res$=esc$+"@":                             REM
DRUCKER RESET
11270                                              REM
11280   DEF FNgn$(x)=esc$+"K"+CHR$(INT(((x/256)-(IN
T(x/256)))*256))+CHR$(INT(x/256))
11290                                              REM
BIT BILDER NORMALE DICHTE
11300   DEF FNgd$(x)=esc$+"L"+CHR$(INT(((x/256)-(IN
T(x/256)))*256))+CHR$(INT(x/256))
11310                                              REM
BIT BILDER DOPPELTE DICHTE
11320                                              REM
11330   komp$=CHR$(15):                            REM
KOMPAKT SCHRIFT
11340   elite$=esc$+"M":                           REM
ELITE   SCHRIFT
11350   RETURN
```

Funktionenplot

Für die Kenner von Mallard-Basic ist es kein Geheimnis; den neuen Joyce-Besitzer verwundert jedoch die Tatsache: der Computer ist grafikfähig, das mitgelieferte Basic aber nicht. Gut – nehmen wir halt die Sprache Logo zwecks der Grafikerzeugung, wozu sie ja konzipiert wurde. Dieses Ansinnen ist aber leider nicht bei jeder Anwendung zweckmäßig. Wer sich aber trotzdem nicht in weitere Unkosten für eine "grafikfähige" Programmiersprache für den Joyce stürzen will, der kann sich mit folgendem Programm weiterhelfen – mit praktischem Nutzen und als Anregung.

Es handelt sich hierbei um einen zunächst "stink" (Verzeihung!) normalen Funktionenplotter. Der einzige Vorteil zu anderen Plottern, die ja käuflich erstanden werden müssen, wenn man sie nicht selbst programmiert und baut, ist, daß sich dieser einfache Plotter mit der mitgelieferten Soft- und Hardware des Joyce realisieren läßt.
Wie oben schon erwähnt, unterstützt Mallard-Basic nicht die Grafikfähigkeit des Joyce, die Sprache Logo aber sehr wohl. Der Funktionenplotter verbindet nun auf einfache Weise die Vorteile der beiden Sprachen:
- Basic eignet sich recht gut zur Verarbeitung von mathematischen Formeln, weil es wesentlich schneller als Logo ist
- Logo ist hingegen grafikfähig, eignet sich aber nicht besonders gut zur Formelverarbeitung, zumal wichtige Funktionen wie etwa die Exponentialfunktion fehlen

Was liegt also ferner, als die zwei zusammenarbeiten zu lassen!

So geht's

Als erstes erstellt man sich eine Diskette mit den für den Plotter nötigen Programmen. Dies sind einmal die zwei Dateien mit den Programmiersprachen BASIC.COM (1. Systemdiskette, Seite 2) und LOGO.COM (2. Systemdiskette, Seite 2) sowie das Programm zur Verarbeitung von Stapeldateien SUBMIT.COM (1. Systemdiskette, Seite 2). Dazu sei auch nochmal auf den Artikel "Startdisketten anders erstellt" aus Heft 6/86, Seite 23 hingewiesen. Dazu werden noch das Basic-Programm PLOT.BAS (Listing 1) und die Stapeldatei PLOT.SUB (Listing 2) nach erfolgter Eingabe gespeichert.
Bevor man das Basic-Programm zwecks Ausprobieren startet, sollte dieses unbedingt gespeichert werden, da das Programm nach dem Ablauf Basic verläßt und zum Betriebssystem zurückkehrt. Das Programm ist dann unwiderruflich "verschwunden", wenn es vorher nicht gespeichert wurde. Für die Erstellung der Stapeldatei sollte der Editor RPED.BAS verwendet werden, dessen Handhabung sich nach Start mit **BASIC RPED** von selbst erklären dürfte.

Der Ablauf

Die Stapeldatei PLOT.SUB übernimmt nach Aufruf durch Eingabe **SUBMIT PLOT** die Ablaufsteuerung. Die erste Zeile bzw. der erste Befehl (basic plot) dieser Datei bewirkt das Laden des Basic-Interpreters und bringt das Programm PLOT.BAS zur Ausführung. Nun hat man die Möglichkeit, die zu plottende Funktion über die Tastatur einzugeben oder aus einer Datei zu lesen, die (irgendwann) erstellt worden ist. Letzteres geschieht immer automatisch, wenn eine Funktion über die Tastatur eingegeben wird; sie wird unter einen anzugebenden Namen mit dem Dateityp .FKT gespeichert. Warum, erfahren Sie etwas später.
Als nächstes wird der Intervall abgefragt, in dem die Funktion dargestellt werden soll, und daraufhin der benötigte Maßstab der Y-Achse berechnet, um eine größtmögliche Auflösung zu erzielen. Als letztes – und das ist eigentlich der Clou der ganzen Sache – generiert das Basic-Programm ein einfaches Logo-Programm und speichert dieses unter dem Namen PLOT.LOG in das Laufwerk M:. Damit ist die Aufgabe dieses Programmes beendet und mit dem Befehl SYSTEM wird wieder zum Betriebssystem zurückgekehrt.
Hier kommt nun der zweite Befehl der Stapeldatei zur Ausführung: logo.com m:plot. Dieser Befehl bewirkt genau das gleiche wie der Befehl der ersten Zeile, jedoch wird hier der Logo-Interpreter geladen und das in M: befindliche Logo-Programm PLOT.SUB ausgeführt. Als Ergebnis ist die "geplottete" Funktion endlich auf dem Monitor zu begutachten. Hat man sich genug daran ergötzt, so kann man durch die Eingabe von **BYE** (was am Bildschirm nicht angezeigt wird) und Drücken der Return-Taste wieder zum Betriebssystem gelangen und das Ganze wiederholen.

Einzelheiten

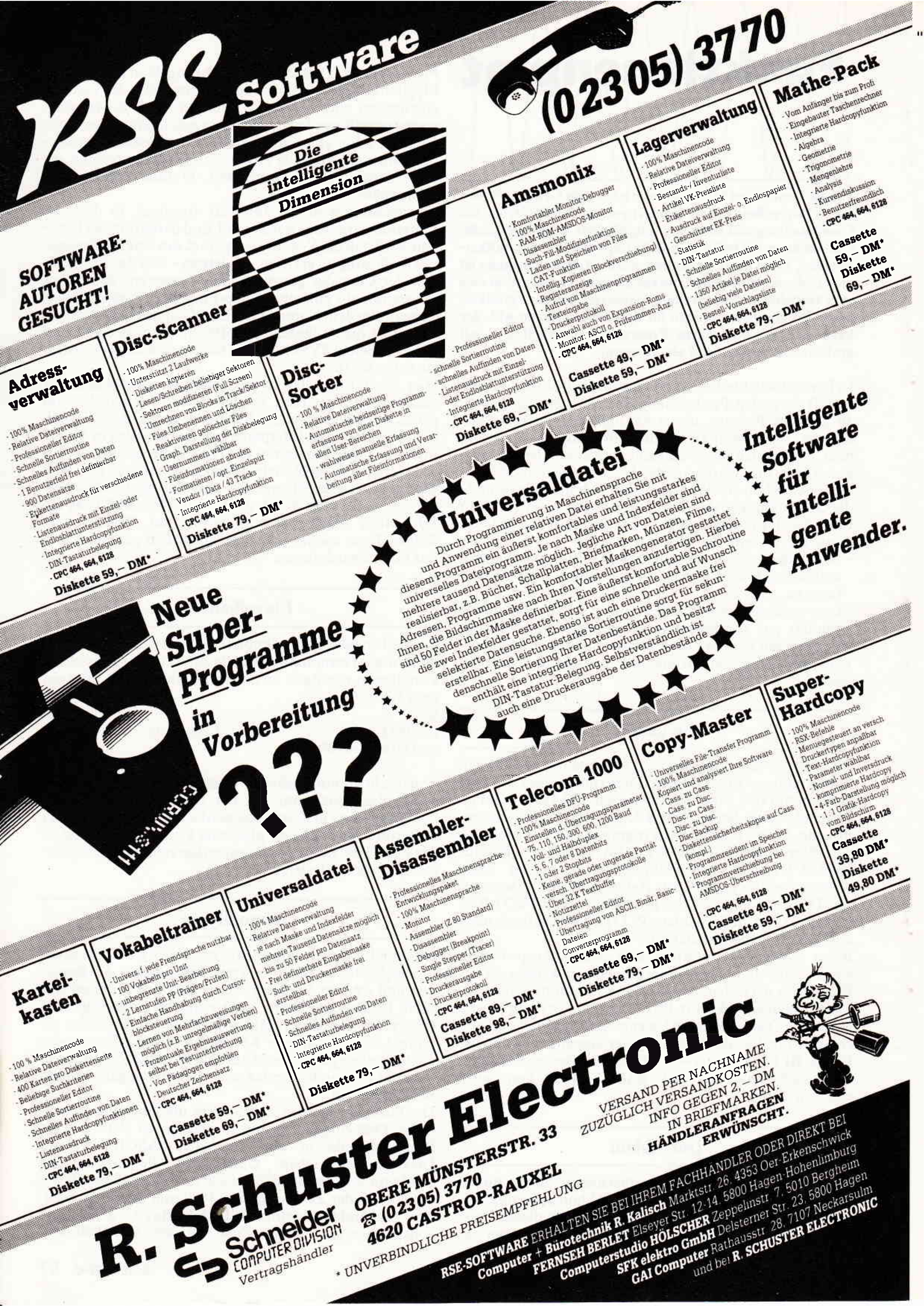
Bei der Eingabe der Funktion (F(x) = ...) ist unbedingt auf die Syntax (Grammatik) und die zur Verfügung gestellten Funktionen von Basic zu achten (s. Handbuch). Hier ein paar Beispiele:

sin(x)	Sinus-Funktion
log10(x)	Logarithmus zur Basis 10
cos(x)*exp(-x/10)	

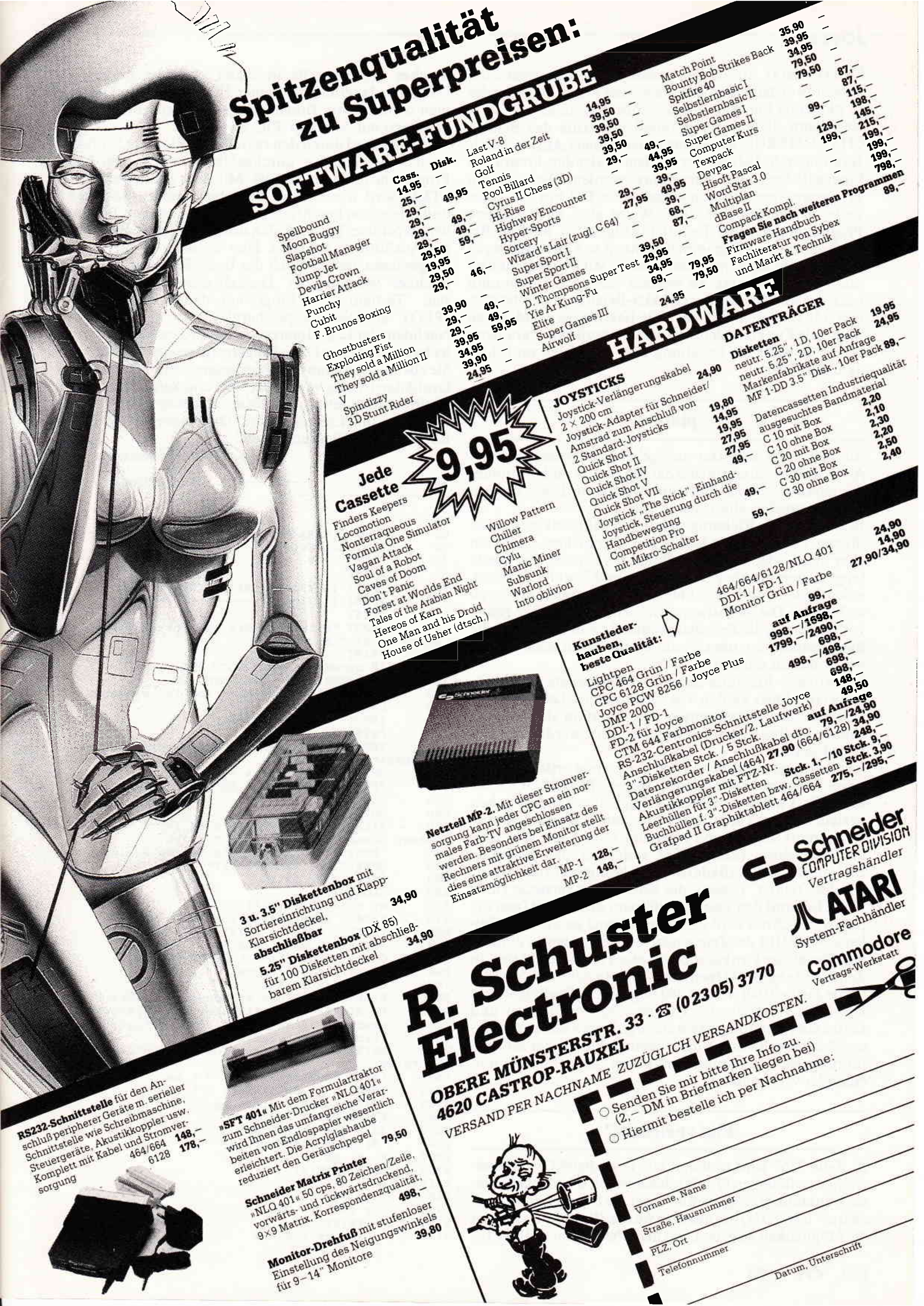
Entspricht die eingegebene Funktion nicht den Regeln, so stoppt das Programm mit der entsprechenden Basic-Fehlermeldung in Zeile 2100, wo die Funktion mit **FNf(x)** aufgerufen wird. Die zu plottende Funktion selbst ist in der Zeile 1000 gespeichert und entsprechend der Fehlermeldung zu analysieren.

selfmodifying

Nun stellt sich die Frage, wie diese bei jeder Funktion unterschiedliche Programmzeile ohne Zutun des Benutzers und Ändern des Programmes durch eben diesen entsteht. Dafür gibt es eine ganz einfache Erklärung. Das Programm modifiziert (ändert) sich selbst. Und dies geschieht ohne raffinierte Kunstgriffe mittels Pokes oder Tastenumbelegung, die eine exakte Kenntnis des Innenlebens des Interpreters verlangen, sondern mit ganz legalen Basic-Befehlen.
Die Funktion wird in Zeile 180 in die Zeichenkettenvariable **Fkt$** eingelesen, und in den Zeilen 200 – 230 als Programmzeile in eine Textdatei (ASCII-File) auf Diskette geschrieben. Dazu wird die eingegebene Funktion, z.B. **sin(x)**, in Zeile 220 noch gemäß der Basic-Syntax mit der Zeilennummer 1000 und dem Schlüsselwort zur Funktionsdefinition **DEF FNf(x)=** ergänzt. Eine vollständige, gespeicherte Zeile wäre also z.B. **1000 DEF**



RSE Software
(02305) 3770
Die intelligente Dimension
SOFTWARE-AUTOREN GESUCHT!
Mathe-Pack
- Vom Anfänger bis zum Profi
- Eingebauter Taschenrechner
- Integrierte Hardcopyfunktion
- Algebra
- Geometrie
- Trigonometrie
- Mengenlehre
- Analysis
- Kurvendiskussion
- Benutzerfreundlich
- CPC 464, 664, 6128
Cassette 59,– DM*
Diskette 69,– DM*
Lagerverwaltung
- 100% Maschinencode
- Relative Dateiverwaltung
- Professioneller Editor
- Bestands-/Inventurliste
- Artikel VK-Preisliste
- Etikettenausdruck
- Ausdruck auf Einzel- o. Endlospapier
- Geschützter EK-Preis
- Statistik
- DIN-Tastatur
- Schnelle Sortierroutine
- Schnelles Auffinden von Daten
- 1350 Artikel je Datei möglich (beliebig viele Dateien)
- Bestell-Vorschlagsliste
- CPC 464, 664, 6128
Diskette 79,– DM*
Amsmonix
- Komfortabler Monitor-Debugger
- 100% Maschinencode
- RAM-ROM-AMSDOS-Monitor
- Disassembler
- Such-Fill-Modifizierfunktion
- Laden und Speichern von Files
- CAT-Funktion
- Intellig. Kopieren (Blockverschiebung)
- Registeranzeige
- Aufruf von Maschinenprogrammen
- Texteingabe
- Druckerprotokoll
- Anwahl auch von Expansion-Roms
- Monitor: ASCII o. Prüfsummen-Anz.
- CPC 464, 664, 6128
Cassette 49,– DM*
Diskette 59,– DM*
Adress-verwaltung
- 100% Maschinencode
- Relative Dateiverwaltung
- Professioneller Editor
- Schnelle Sortierroutine
- Schnelles Auffinden von Daten
- 1 Benutzerfeld frei definierbar
- 900 Datensätze
- Etikettenausdruck für verschiedene Formate
- Listenausdruck mit Einzel- oder Endlosblattunterstützung
- Integrierte Hardcopyfunktion
- DIN-Tastaturbelegung
- CPC 464, 664, 6128
Diskette 59,– DM*
Disc-Scanner
- 100% Maschinencode
- Unterstützt 2 Laufwerke
- Disketten kopieren
- Lesen/Schreiben beliebiger Sektoren
- Sektoren modifizieren (Full Screen)
- Umrechnen von Blocks in Track/Sektor
- Files Umbenennen und Löschen
- Reaktivieren gelöschter Files
- Graph. Darstellung der Diskbelegung
- Usernummern wählbar
- Fileinformationen abrufen
- Formatieren / opt. Einzelspur Vendor / Data / 43 Tracks
- Integrierte Hardcopyfunktion
- CPC 464, 664, 6128
Diskette 79,– DM*
Disc-Sorter
- 100 % Maschinencode
- Relative Dateiverwaltung
- Automatische beidseitige Programmerfassung von einer Diskette in allen User-Bereichen
- wahlweise manuelle Erfassung
- Automatische Erfassung und Verarbeitung aller Fileinformationen
- Professioneller Editor
- schnelle Sortierroutine
- schnelles Auffinden von Daten
- Listenausdruck mit Einzel- oder Endlosblattunterstützung
- Integrierte Hardcopyfunktion
- CPC 464, 664, 6128
Diskette 69,– DM*
Universaldatei
Durch Programmierung in Maschinensprache und Anwendung einer relativen Datei erhalten Sie mit diesem Programm ein äußerst komfortables und leistungsstarkes universelles Dateiprogramm. Je nach Maske und Indexfelder sind mehrere tausend Datensätze möglich. Jegliche Art von Dateien sind realisierbar, z.B. Bücher, Schallplatten, Briefmarken, Münzen, Filme, Adressen, Programme usw. Ein komfortabler Maskengenerator gestattet Ihnen, die Bildschirmmaske nach Ihren Vorstellungen anzufertigen. Hierbei sind 50 Felder in der Maske definierbar. Eine äußerst komfortable Suchroutine die zwei Indexfelder gestattet, sorgt für eine schnelle und auf Wunsch selektierte Datensuche. Ebenso ist auch eine Druckermaske frei erstellbar. Eine leistungsstarke Sortierroutine sorgt für sekundenschnelle Sortierung Ihrer Datenbestände. Das Programm enthält eine integrierte Hardcopyfunktion und besitzt DIN-Tastatur-Belegung. Selbstverständlich ist auch eine Druckerausgabe der Datenbestände
Intelligente Software für intelligente Anwender.
Neue Super-Programme in Vorbereitung ???
Super-Hardcopy
- 100% Maschinencode
- RSX-Befehle
- Menuegesteuert an versch. Druckertypen anpaßbar
- Text-Hardcopyfunktion
- Parameter wählbar
- Normal- und Inversdruck
- komprimierte Hardcopy
- 4-Farb-Darstellung möglich
- 1:1 Grafik-Hardcopy vom Bildschirm
- CPC 464, 664, 6128
Cassette 39,80 DM*
Diskette 49,80 DM*
Copy-Master
- Universelles File-Transfer Programm
- 100% Maschinencode
Kopiert und analysiert Ihre Software
- Cass. zu Cass.
- Cass. zu Disc
- Disc zu Cass.
- Disc zu Disc
- Disc Backup
- Diskettensicherheitskopie auf Cass. (kompl.)
- Programmresident im Speicher
- Integrierte Hardcopyfunktion
- Programmverschiebung bei AMSDOS-Überschreibung
- CPC 464, 664, 6128
Cassette 49,– DM*
Diskette 59,– DM*
Telecom 1000
- Professionelles DFÜ-Programm
- 100% Maschinencode
- Einstellen d. Übertragungsparameter
- 75, 110, 150, 300, 600, 1200 Baud
- Voll- und Halbduplex
- 5, 6, 7 oder 8 Datenbits
- 1 oder 2 Stopbits
- Keine, gerade oder ungerade Parität
- versch. Übertragungsprotokolle
- Über 32 K Textbuffer
- Notizzettel
- Professioneller Editor
- Übertragung von ASCII, Binär, Basic-Dateien
- Converterprogramm
- CPC 464, 664, 6128
Cassette 69,– DM*
Diskette 79,– DM*
Assembler-Disassembler
- Professionelles Maschinensprache-Entwicklungspaket
- 100% Maschinensprache
- Monitor
- Assembler (Z 80 Standard)
- Disassembler
- Debugger (Breakpoint)
- Single Stepper (Tracer)
- Professioneller Editor
- Druckerausgabe
- Druckerprotokoll
- CPC 464, 664, 6128
Cassette 89,– DM*
Diskette 98,– DM*
Universaldatei
- 100% Maschinencode
- Relative Dateiverwaltung
- je nach Maske und Indexfelder mehrere Tausend Datensätze möglich
- bis zu 50 Felder pro Datensatz
- Frei definierbare Eingabemaske
- Such- und Druckermaske frei erstellbar
- Professioneller Editor
- Schnelle Sortierroutine
- Schnelles Auffinden von Daten
- DIN-Tastaturbelegung
- Integrierte Hardcopyfunktion
- CPC 464, 664, 6128
Diskette 79,– DM*
Vokabeltrainer
- Univers. f. jede Fremdsprache nutzbar
- 100 Vokabeln pro Unit
- unbegrenzte Unit-Bearbeitung
- 2 Lernstufen PP (Prägen/Prüfen)
- Einfache Handhabung durch Cursorblocksteuerung
- Lernen von Mehrfachzuweisungen möglich (z.B. unregelmäßige Verben)
- Prozentuale Ergebnisauswertung, selbst bei Testunterbrechung
- Von Pädagogen empfohlen
- Deutscher Zeichensatz
- CPC 464, 664, 6128
Cassette 59,– DM*
Diskette 69,– DM*
Kartei-kasten
- 100 % Maschinencode
- Relative Dateiverwaltung
- 400 Karten pro Diskettenseite
- Beliebige Suchkriterien
- Professioneller Editor
- Schnelle Sortierroutine
- Schnelles Auffinden von Daten
- Integrierte Hardcopyfunktionen
- Listenausdruck
- DIN-Tastaturbelegung
- CPC 464, 664, 6128
Diskette 79,– DM*
R. Schuster Electronic
OBERE MÜNSTERSTR. 33
(02305) 3770
4620 CASTROP-RAUXEL
Schneider COMPUTER DIVISION Vertragshändler
* UNVERBINDLICHE PREISEMPFEHLUNG
VERSAND PER NACHNAME ZUZÜGLICH VERSANDKOSTEN. INFO GEGEN 2,– DM IN BRIEFMARKEN.
HÄNDLERANFRAGEN ERWÜNSCHT.
RSE-SOFTWARE ERHALTEN SIE BEI IHREM FACHHÄNDLER ODER DIREKT BEI
Computer + Bürotechnik R. Kalisch Marktstr. 26, 4353 Oer-Erkenschwick
FERNSEH BERLET Elseyer Str. 12-14, 5800 Hagen-Hohenlimburg
Computerstudio HÖLSCHER Zeppelinstr. 7, 5010 Bergheim
SFK elektro GmbH Delsterner Str. 23, 5800 Hagen
GAI Computer Rathausstr. 28, 7107 Neckarsulm
und bei R. SCHUSTER ELECTRONIC



Spitzenqualität zu Superpreisen:
SOFTWARE-FUNDGRUBE
Cass. Disk.
Spellbound 14.95 –
Moon Buggy 25,– 49,95
Slapshot 29,– –
Football Manager 29,– 49,–
Jump-Jet 29,– 49,–
Devils Crown 29,– 59,–
Harrier Attack 29,50 –
Punchy 19,95 –
Cubit 29,50 46,–
F. Brunos Boxing 29,– –
Ghostbusters 39,90 49,–
Exploding Fist 29,– 49,–
They sold a Million 29,– 59,95
They sold a Million II 39,95 –
V 34,95 –
Spindizzy 39,90 –
3D Stunt Rider 24,95 –
Last V-8 14,95 –
Roland in der Zeit 39,50 –
Golf 39,50 –
Tennis 39,50 –
Pool Billard 39,50 49,–
Cyrus II Chess (3D) 29,– 44,95
Hi-Rise – 39,95
Highway Encounter – 39,–
Hyper-Sports 29,– 49,95
Sorcery 27,95 39,–
Wizard's Lair (zugl. C 64) – 68,–
Super Sport I – 87,–
Super Sport II – –
Winter Games 39,50 –
D. Thompsons Super Test 29,95 –
Yie Ar Kung-Fu 34,95 79,95
Elite 69,– 79,50
Super Games III 24,95
Airwolf
Match Point 35,90 –
Bounty Bob Strikes Back 39,95 –
Spitfire 40 34,95 –
Selbstlernbasic I 79,50 –
Selbstlernbasic II 79,50 87,–
Super Games I – 87,–
Super Games II – 115,–
Computer Kurs 99,– 198,–
Texpack – 145,–
Devpac 129,– 215,–
Hisoft Pascal 199,– 199,–
Word Star 3.0 – 199,–
Multiplan – 199,–
dBase II – 798,–
Compack Kompl. 89,–
Fragen Sie nach weiteren Programmen
Firmware Handbuch
Fachliteratur von Sybex
und Markt & Technik
HARDWARE
JOYSTICKS
Joystick-Verlängerungskabel 2 × 200 cm 24,90
Joystick-Adapter für Schneider/Amstrad zum Anschluß von 2 Standard-Joysticks 19,80
Quick Shot I 14,95
Quick Shot II 19,95
Quick Shot IV 27,95
Quick Shot V 27,95
Quick Shot VII 49,–
Joystick „The Stick", Einhand-Joystick, Steuerung durch die Handbewegung 49,–
Competition Pro mit Mikro-Schalter 59,–
DATENTRÄGER
Disketten
neutr. 5.25", 1D, 10er Pack 19,95
neutr. 5.25", 2D, 10er Pack 24,95
Markenfabrikate auf Anfrage
MF 1-DD 3.5" Disk., 10er Pack 89,–
Datencassetten Industriequalität ausgesuchtes Bandmaterial
C 10 mit Box 2,20
C 10 ohne Box 2,10
C 20 mit Box 2,30
C 20 ohne Box 2,20
C 30 mit Box 2,50
C 30 ohne Box 2,40
Jede Cassette 9,95
Finders Keepers
Locomotion
Nonterraqueous
Formula One Simulator
Vagan Attack
Soul of a Robot
Caves of Doom
Don't Panic
Forest at Worlds End
Tales of the Arabian Night
Hereos of Karn
One Man and his Droid
House of Usher (dtsch.)
Willow Pattern
Chiller
Chimera
Cylu
Manic Miner
Subsunk
Warlord
Into oblivion
Kunstlederhauben, beste Qualität:
464/664/6128/NLQ 401 24,90
DDI-1 / FD-1 14,90
Monitor Grün / Farbe 27,90/34,90
Lightpen 99,–
CPC 464 Grün / Farbe auf Anfrage
CPC 6128 Grün / Farbe 998,–/1698,–
Joyce PCW 8256 / Joyce Plus 1799,–/2490,–
DMP 2000 698,–
DDI-1 / FD-1 498,–/498,–
FD-2 für Joyce 698,–
CTM 644 Farbmonitor 698,–
RS-232-Centronics-Schnittstelle Joyce 148,–
Anschlußkabel (Drucker/2. Laufwerk) 49,50
3"-Disketten Stck. / 5 Stck. auf Anfrage
Datenrekorder / Anschlußkabel dto. 79,–/24,90
Verlängerungskabel (464) 27,90 (664/6128) 34,90
Akustikkoppler mit FTZ-Nr. 248,–
Leerhüllen für 3"-Disketten Stck. 1,–/10 Stck. 9,–
Buchhüllen f. 3"-Disketten bzw. Cassetten Stck. 3,90
Grafpad II Graphiktablett 464/664 275,–/295,–
Netzteil MP-2. Mit dieser Stromversorgung kann jeder CPC an ein normales Farb-TV angeschlossen werden. Besonders bei Einsatz des Rechners mit grünem Monitor stellt dies eine attraktive Erweiterung der Einsatzmöglichkeit dar.
MP-1 128,–
MP-2 148,–
3 u. 3.5" Diskettenbox mit Sortiereinrichtung und Klappsichtdeckel, Klarsichtdeckel, abschließbar 34,90
5.25" Diskettenbox (DX 85) für 100 Disketten mit abschließbarem Klarsichtdeckel 34,90
Schneider COMPUTER DIVISION Vertragshändler
ATARI System-Fachhändler
Commodore Vertrags-Werkstatt
R. Schuster Electronic
OBERE MÜNSTERSTR. 33 · ☎ (02305) 3770
4620 CASTROP-RAUXEL
VERSAND PER NACHNAME ZUZÜGLICH VERSANDKOSTEN.
RS232-Schnittstelle für den Anschluß peripherer Geräte m. serieller Schnittstelle wie Schreibmaschine, Steuergeräte, Akustikkoppler usw. Komplett mit Kabel und Stromversorgung 464/664 148,– 6128 178,–
»SFT 401« Mit dem Formulartraktor zum Schneider-Drucker »NLQ 401« wird Ihnen das umfangreiche Verarbeiten von Endlospapier wesentlich erleichtert. Die Acrylglashaube reduziert den Geräuschpegel 79,50
Schneider Matrix Printer »NLQ 401« 50 cps, 80 Zeichen/Zeile, vorwärts- und rückwärtsdruckend, 9×9 Matrix, Korrespondenzqualität, 498,–
Monitor-Drehfuß mit stufenloser Einstellung des Neigungswinkels für 9–14" Monitore 39,80
Senden Sie mir bitte Ihre Info zu. (2,– DM in Briefmarken liegen bei)
Hiermit bestelle ich per Nachnahme:
Vorname, Name
Straße, Hausnummer
PLZ, Ort
Telefonnummer
Datum, Unterschrift

FNf(x)=sin(x). Aus diesem Grund können Sie eben auch auf schon erstellte Funktionen bei der Frage "Existiert die zu plottende Funktion schon?" zurückgreifen.
Nun kann dieser Programmteil mit Hilfe des Befehls **CHAIN MERGE dateiname,zeilennummer,ALL** (s. auch Handbuch, S. 184) wieder zu dem laufenden Programm hinzugeladen und ausgeführt werden. Bei diesem Programm besteht der nachgeladene Teil nur aus einer Zeile; es können auf diese Weise aber auch größere Programmteile nach Bedarf hinzugeladen werden. Bei umfangreichen Programmen können so Überlagerungen (Overlays) in Basic realisiert werden. Mit der "zeilennummer" wird bestimmt, an welcher Stelle das Programm nach der Ausführung des CHAIN-Befehls fortgesetzt werden soll. In diesem Fall bei Zeile 1000, da die geladene, zu plottende Funktion definiert werden muß. Der Parameter "ALL" bewirkt die Erhaltung aller Variablen und ihrer aktuellen Werte.

plotten

Aus dem vom Benutzer angegebenen Plotintervall bzw. Abzissenbereich ergibt sich das Maximum und Minimum der darzustellenden Funktion, welche in dieser Version folgendermaßen abgetastet werden:
In horizontaler Richtung stehen 720 Bildpunkte zur Verfügung. Um eine gute Auflösung zu erreichen, muß man aber nicht jeden einzelnen Punkt berechnen, was relativ lange dauert bis die Zeichnung einmal "steht". Es reicht, jeden dritten Punkt zu nehmen und diese mittels Linien zu verbinden. Die Abtastwerte werden, da Speicher genug vorhanden ist, die Zeit aber eventuell nicht, in die Tabelle **ftab()** gespeichert, um sie zum Plotten nicht noch einmal berechnen zu müssen.
In vertikaler Richtung stehen 248 Bildpunkte zur Verfügung (von Logo werden 496 vorgetäuscht). Daraus ergibt sich mit den Extremwerten der Funktion der entsprechende Maßstab und die Funktionswerte werden entsprechend skaliert.
Jetzt wird das eigentliche Logo-Programm erstellt und als Logo-Programmdatei mit dem Namen PLOT.LOG in die RAM-Disk M: geschrieben (die folgenden Zeilenangaben beziehen sich auf das dieses Programm erzeugende Basic-Programm). Das Logo-Programm initialisiert als erstes den Bildschirm (Zeile 2280), indem es den gesamten Bildschirm zur Grafikdarstellung bestimmt ('full screen', fs), ihn "reinigt" (clean), die Schildkröte versteckt (hide turtle, hd) und den Zeichenstift (pen) deaktiviert (pen up, pu). Als nächstes wird das Achsenkreuz gezeichnet (Zeilen 2290 - 2310), der Zeichenstift auf den ersten zu plottenden Punkt der Funktion positioniert (Zeile 2320) und in Zeile 2330 abgesenkt (pen down, pd). In der FOR-Schleife (Zeile 2340 - 2360) werden 240 Logo-SETPOS-Befehle mit den beim Abtasten der Extremwerte berechneten und skalierten Funktionswerten erzeugt. Durch den abgesenkten Zeichenstift werden die einzelnen Punkte bei der Ausführung des Logo-Programms durch "kurze" Linien miteinander verbunden.

Plot speichern?

In Zeile 2365 wird optional ein Logo-Befehl zum Speichern des aktuellen Grafikbildschirmes, d.h. der geplotteten Funktion, erzeugt, wenn vorher die Frage "Grafik speichern?" mit "j" (ja) beantwortet wurde. Warum? Um Ihnen die Möglichkeit zu geben, mit dem Programm COPYPIC, welches Sie ebenfalls in dieser Ausgabe finden, die Funktion dauerhaft zu Papier zu bringen. Als Dateinamen wird der von Ihnen angegebene verwendet, jedoch von Logo mit dem Typ .PIC versehen.
Als letztes wird durch den recycle-Befehl (Zeile 2370) eine Speicherbereinigung durchgeführt, womit das Logo-Programm fertig erzeugt ist. Mit dem Basic-Befehl **SYSTEM** wird wieder zum Betriebssystem zurückgekehrt und, wie schon beschrieben, durch den zweiten Befehl in der Stapeldatei Logo geladen und das erzeugte Programm zur Ausführung gebracht. Hier sei noch einmal darauf hingewiesen, daß dadurch das Basic-Programm aus dem Rechner "verschwindet". Deshalb unbedingt vor dem ersten "Testlauf" das Programm dauerhaft mit SAVE "PLOT" auf Diskette speichern!
Nachdem der Logo-Interpreter das Programm abgearbeitet hat, wartet er auf eine Eingabe Ihrerseits. Dies können Sie aber nicht erkennen, da der gesamte Bildschirm für die Grafikdarstellung verwendet wird und kein Platz für den Dialog vorhanden ist. Sie müssen also, wie oben schon beschrieben, Ihre Eingabe "blind" tätigen, wie z.B. **bye** für die Rückkehr zum Betriebssystem.

(Karl Rudolf Meissner/MC)

```
10        '------------------- FUNKTIONENPLOT -----
-------------------
20        '
40        '
50        cls$ = CHR$(27) + "E" + CHR$(27) + "H"
60        DIM ftab(240)
70        PRINT cls$
80        PRINT
90        PRINT TAB(15);"FUNKTIONENPLOT"
100       PRINT TAB(15);"--------------"
110       PRINT
120       INPUT "existiert die zu plottende Funktion
 schon (j/n) ";antw$
130       PRINT
140       IF antw$="j" THEN GOTO 250
150       IF antw$<>"n" THEN GOTO 120
160       PRINT "bitte Funktion von 'x' eingeben:"
170       PRINT
180       INPUT "F(x) = ",Fkt$
190       PRINT
200       INPUT "unter welchem Namen ablegen (max. 8
 Zeichen)";Fktn$
210       OPEN "O",1,fktn$ + ".FKT"
220       PRINT # 1,"1000     DEF FNf(x)=";fkt$
230       CLOSE 1
240       GOTO 500
250       FILES "*.FKT":PRINT:PRINT:INPUT "bitte Dat
einamen (ohne Typ) angeben : ",fktn$
260       'Funktion zunächst als String einlesen
270       OPEN "I",1,fktn$+".FKT"
280       INPUT # 1, fkt$
290       CLOSE 1
300       sp = INSTR(fkt$,"f(x)")
310       fkt$ = RIGHT$(fkt$,(LEN(fkt$)-sp+1))
320       PRINT : PRINT : PRINT fkt$ : PRINT
499       'Funktion dazuladen
500       CHAIN MERGE fktn$+".FKT",1000,ALL
999       'Funktion definieren
1000      DEF FNf(x)=SIN(x)
2000     'Plot als PICture speichern für COPY-PIC?
2005      PRINT:INPUT "Grafik speichern ";antw$
2010      savflg=0:IF antw$="j" THEN savflg=1
2015      'Plotbereeich bestimmen:
2020      PRINT:PRINT "bitte Plot-Intervall angeben:
" : PRINT
2030      INPUT "Anfangs-X-Wert = ",xa
2040      INPUT "     End-X-Wert = ",xe
2060      'Maßstab in x-Richtung
2065      PRINT:PRINT"rechne:"
2070      dx = (xe-xa)/720*3
2080      'Minimum und Maximum
2090      x  =  xa
2100      mi =  FNf(x)
2110      ma = mi
2120      ftab(0) = mi
2130      FOR i%=1 TO 240
2135         PRINT".";
2140         x = xa + dx*i%
```

Bitte senden Sie mir folgende Artikel aus dem ☐ Mailorder Angebot ☐ Clubangebot

☐ Ich möchte weitere Informationen über den Data Media Computer-Club, ehe ich mich zu einer Mitgliedschaft entschließe (2,– DM in Briefmarken habe ich beigelegt). Mein Name und meine Anschrift lautet:

____

____

____

____

____

Datum Unterschrift (bei Minderjährigen des gesetzl. Vertreters)

data media gmbh

Bitte ausreichend frankieren

Data Media GmbH
– Mailorder/Computerclub -
Ruhralle 55

4600 Dortmund

Schneider CPC INTERNATIONAL

»Klein-Anzeigen-Markt«

Absender: *(Bitte genaue Anschrift angeben!)*

Name

Vorname

Firma

Straße/Nr./Postfach

PLZ/Ort

Bitte ausreichend frankieren

Antwortkarte

Schneider CPC International
Postfach 250

3440 Eschwege



Bestellkarte »LESERSERVICE«

Absender: *(Bitte genaue Anschrift angeben!)*

Name

Vorname

Firma

Straße/Nr./Postfach

PLZ/Ort

Bitte ausreichend frankieren

Antwortkarte

Schneider CPC International
Postfach 250

3440 Eschwege

Schneider CPC INTERNATIONAL

CPC-Platinen- und Hardware-Service

Absender: *(Bitte genaue Anschrift angeben!)*

Name

Vorname

Firma

Straße/Nr./Postfach

PLZ/Ort

Bitte ausreichend frankieren

Antwortkarte

Schneider CPC International
Postfach 250

3440 Eschwege

CPC-Platinen- und Hardware-Service

Ich bestelle hiermit gemäß Ihrem Angebot aus "Schneider CPC International", Ausgabe 7/86, Schneiderware # 2:

.... Stck.	Basisplatine, beidseitig beschichtet, durchkontaktiert	**24,90 DM**
.... Stck.	Basisplatine kompl. bestückt u. geprüft, funktionsfertig	**62,90 DM**
.... Stck.	Centronics-Platine, einseitig beschichtet	**17,90 DM**
.... Stck.	Centronics-Platine komplett bestückt und geprüft, funktionsfertig	**79,90 DM**
.... Stck.	Verbindungskabel zwischen Basisplatine und CPC 464/664	**39,50 DM**
.... Stck.	Verbindungskabel zwischen Basisplatine u. CPC 6128	**45,90 DM**

Ausgabe 8/86, Schneiderware # 3

.... Stck.	V/24 Platine beidseitig beschichtet, durchkontaktiert	**29,80 DM**
.... Stck.	V/24 Platine kompl. bestückt und geprüft, funktionsfertig	**139,90 DM**

Betrag **DM**

+ Porto/Verpackung
(Porto/Verp. Inland 5,– DM)
(Porto/Verp. Ausland 8,– DM) DM

Gesamtbetrag DM

☐ Diesen Betrag bezahle ich mittels des beigefügten Verrechnungsschecks.
☐ Ich bitte um Lieferung per Nachnahme (nur innerhalb der BRD).
Bei Nachnahme kommt zum o.g. Betrag noch die Nachnahmegebühr hinzu.

Datum Unterschrift (bei Minderjährigen des gesetzl. Vertreters)

»Kleinanzeigen-Markt«

Private Anzeigen: Nur DM 5,- je angefangene Zeile, inkl. gesetzlicher Mehrwertsteuer.
Geschäftliche Empfehlungen: DM 7,- je angefangene Zeile, zzgl. gesetzlicher Mehrwertsteuer.
Kreuzen Sie bitte an, in welche Rubrik (s. Karte) Ihre Anzeige gehört, schreiben Sie Ihren Text in die Karte (jedes Kästchen = ein Zeichen; Buchstabe, Satzzeichen oder Wortzwischenraum).
Absender und Unterschrift auf die Karte und heraustrennen.
Achtung! Der Abdruck erfolgt **nur** gegen Vorkasse (Verrechnungsscheck).
Bitte veröffentlichen Sie meine Anzeige in der nächsterreichbaren **„Schneider CPC International"** für
☐ private Zwecke ☐ gewerbliche Zwecke (gewerbliche Anzeigen werden mit **G** gekennzeichnet)
Das ist der Text:

☐ Die Anzeige soll als Chiffre-Anzeige erscheinen (nur möglich bei Privat-Anzeige). Chiffre-Gebühr: DM 10,- inkl MwSt. zzgl. zum Anzeigenpreis.

Bei Angeboten:
Ich bestätige, daß ich alle Rechte an den angebotenen Sachen besitze.

In dieser Rubrik: ☐ **Suche** ☐ **Tausch** ☐ **Stellenmarkt/freie Mitarbeit**
☐ **Biete an** ☐ Hardware ☐ **Geschäftsverbindungen**
☐ Hardware ☐ Software ☐ **Verschiedenes**
☐ Software

Datum: Unterschrift (bei Minderjährigen des ges. Vertreters)

»Leserservice«

Ich mache von Ihrem Angebot Gebrauch und bestelle hiermit:

☐ **GAC** – das Traumtool für den Abenteuerfreund. Mit komplettem deutschen Handbuch. für Schneider CPC 464/664/6128
Stück .... ☐ Kass. à 85,– DM
Stück .... ☐ Disk. à 95,– DM

☐ **"Goldene 7" -1-** – die Qualitätssoftware von Lesern für Leser. Mit sieben Aktions- und Abenteuer-Spielen. für Schneider CPC 464/664/6128
Stück .... ☐ Kass. à 20,– DM
Stück .... ☐ Disk. à 30,– DM

☐ **"Goldene 7" -2-** – die Qualitätssoftware von Lesern für Leser. Jetzt weitere sieben Aktions-, Abenteuer- und Geschicklichkeits-Spiele. für Schneider CPC 464/664/6128
Stück .... ☐ Kass. à 20,– DM
Stück .... ☐ Disk. à 30,– DM

Porto/Verpackung 3,– DM

Gesamtbetrag

☐ Den vorstehenden Betrag bezahle ich mittels des beigefügten Verrechnungsschecks.
☐ Ich bitte um Lieferung per Nachnahme zuzüglich der Nachnahmegebühr (nur innerhalb der BRD).

Datum Unterschrift (bei Minderjährigen Unterschrift des gesetzl. Vertreters)

»Data Media Computerclub«
Beitrittserklärung

Ich möchte von den günstigen Einkaufsangeboten des Data Media Computerclubs Gebrauch machen und beantrage hiermit die Mitgliedschaft ab 1.5.1986 zunächst für ein Jahr. Die Mitgliedschaft soll sich um jeweils ein weiteres Jahr verlängern, wenn ich nicht 3 Monate vor Ablauf des Zeitraumes kündige.
Ich kann ab sofort aus dem Club-Angebot einkaufen und alle weiteren Vorteile des Clubs nutzen. Der Clubausweis wird mir nach Zahlung des Jahresbeitrages von 60,– DM zugesandt.

Vor- und Zuname:

Straße/Haus-Nr.:

Postleitzahl: Wohnort:

Beruf:

Geburtsdatum: Telefon:

Datum 1. Unterschrift (bei Minderjährigen: Vor- und Zuname d. ges. Vertreters)

Ich kann eine Woche nach Erhalt der Mitgliedsbestätigung diesen Beitritt schriftlich widerrufen. Der Widerruf ist zu richten an: Data Media GmbH, Computerclub, Ruhrallee 55, 4600 Dortmund. Ich bestätige durch meine 2. Unterschrift, daß ich auf dieses Widerrufsrecht aufmerksam gemacht worden bin:

Datum 2. Unterschrift (bei Minderjährigen: Vor- und Zuname d. ges. Vertreters)

```
2150        y = FNf(x)
2160        ftab(i%) = y
2170        IF y>ma THEN ma = y  :  GOTO 2190
2180        IF y<mi THEN mi = y
2190     NEXT i%:PRINT:PRINT
2200     PRINT "           MINIMUM = ";mi
2210     PRINT "           MAXIMUM = ";ma
2220      dy = 496/(ma-mi)
2230     'Skalieren
2240     FOR i%=0 TO 240
2250       ftab(i%) = ROUND((ftab(i%)-mi)*dy-248,1)
2260     NEXT i%
2265     PRINT:PRINT"erstelle Logo-Programm:"
2270     OPEN "O",1,"M:PLOT.LOG"
2275     'LOGO -Bildschirm initialisieren
2280      PRINT # 1,"fs clean ht pu"
2285     'Achsenkreuz zeichnen
2290      PRINT # 1,"setpos Ä-360 250ü pd"
2300       PRINT # 1,"setpos Ä-360 -250ü"
2310       PRINT # 1,"setpos Ä 360 -250ü"
2320      PRINT # 1,"pu setpos Ä -360 ";ftab(0);" ü
"
2330       PRINT # 1,"pd"
2340      FOR i%=0 TO 240
2345        PRINT".";
2350        PRINT # 1,"setpos Ä";i%*3-360;" ";ftab(i
%);"ü "
2360      NEXT i%
2365       IF savflg THEN PRINT # 1,"savepic ";CHR$(
34);fktn$
2370       PRINT # 1,"recycle"
2380      CLOSE 1
2390 SYSTEM

basic a:plot
logo.com m:plot
```

Grafik auf dem JOYCE:

mit Grafik- und Sonderzeichen

Dieser Artikel ist als eine weitere Anregung für die JOYCE-Besitzer gedacht, um ihrem Rechner ohne zusätzliche Ausgaben für Software etwas Grafik zu entlocken. Zwar kann man mit Mallard-Basic nicht die hochauflösende Grafik benutzen, wie es mit Logo möglich ist – der Zeichenvorrat des JOYCE umfaßt aber außer den gewöhnlichen alphanumerischen Zeichen eine große Anzahl von Grafikzeichen wie Striche, Ecken etc. in unterschiedlichen Formen sowie ein paar Sonderzeichen, die auch "grafisch" gedeutet werden können. Mit diesen Zeichen kann man durchaus anschauliche "Bilder" darstellen, wie anhand der Kirche in Bild 1 zu sehen ist.

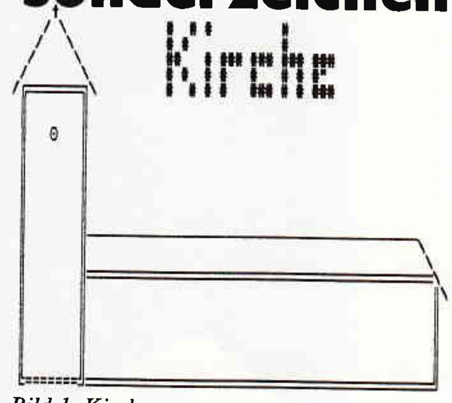


Bild 1: Kirche

Betrachtet man sich den angeblich "vollständigen" Zeichensatz des JOYCE im Handbuch, Anhang I.4, und versucht, einige der bei der Kirche verwendeten Zeichen wie die Doppelstriche zu finden, so wird man dieses vergeblich tun. Lediglich die in der Tabelle als Erweiterungszeichen zusammengefaßten und nicht im Detail aufgeführten Zeichencodes 128 – 159 lassen die Vermutung aufkommen, daß sich dort noch einige interessante Zeichen verbergen könnten. Der unermüdliche Forscher wird sich also nicht scheuen, mit Hilfe des folgenden – oder eines ähnlichen Miniprogramms die "eventuell" mit den Codes verbundenen Zeichen auf dem Bildschirm etwas näher zu betrachten:

FOR i=128 TO 159:PRINT i, CHR$(i):NEXT

Und wie man anhand der Ausgabe zweifelsfrei erkennt, bietet der Zeichensatz des JOYCE, außer den dem Handbuch zu entnehmenden, schon recht vielfältigen Zeichen, noch weitere nützliche Zeichen und Symbole. Mit diesen lassen sich z.B. die von LocoScript bekannten Rahmen für Menues oder Eingabefelder "zeichnen" - oder aber "grobe" Grafiken wie die Kirche erstellen.

Grafikzeichen-Tabelle:

Damit Sie nicht immer mit Hilfe des obigen Einzeilers auf die Suche nach

3''-DISKETTEN

Qualität made in Germany

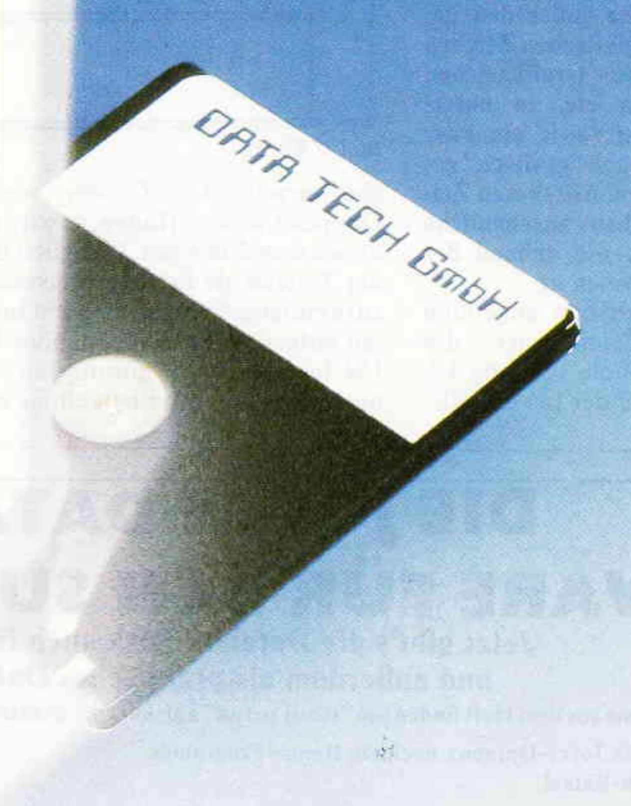
Mit der neuen

3''-Diskette DDM 3-2

hat der Engpaß ein Ende.

100%-ige Datensicherheit, jede Diskette ist geprüft. Double sided, double Face. Medium (Datenträger), beste Qualität zu vernünftigen Preisen.

Data Tech Disketten gibt es jetzt im guten Fachhandel.



Händleranfragen erwünscht, Info über:

DATA TECH GMBH

Ruhrallee 55, 4600 Dortmund

Telefon: 02 31/12 50 74, Telex: 822 7839 datd

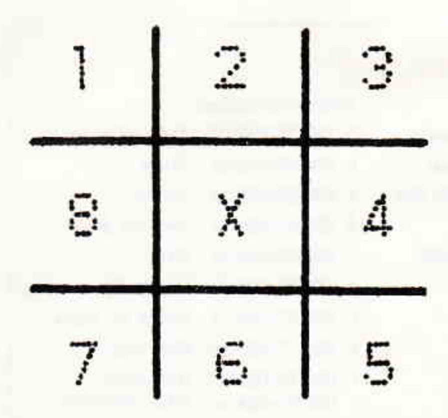


Bild 2: Werte der möglichen Richtungen

dem richtigen Zeichencode gehen müssen, haben wir eine Tabelle mit den im Handbuch nicht aufgeführten Grafikzeichen erstellt. Bei dieser sind auch einige "normale" bzw. landesspezifische Sonderzeichen zweckentfremdet worden und dank ein wenig Phantasie zu einer grafischen bzw. symbolischen Bedeutung gekommen. Die Zeichen wurden in der Tabelle entsprechend ihres Types in Gruppen gegliedert. In der ersten Spalte ist zur Orientierung jeweils die Bildschirmdarstellung des Zeichens zu sehen, die aber aus drucktechnischen Gründen ein wenig von der Darstellung des Zeichens auf Ihrem Rechner differieren kann. In der zweiten Spalte befindet sich der Zeichencode in Form der Basic-CHR$-Funktion, die zur Ausgabe dieser Zeichen auf dem Bildschirm notwendig ist, da die Zeichen nicht durch einen einfachen Tastendruck zu erreichen sind. Zum Beispiel wird der senkrechte Strich mit PRINT CHR$(149) oder das Dreieck mit PRINT CHR$(27);CHR$(3) ausgegeben.

die Kirche im Dorf...

Als Demonstration eines mit den Grafikzeichen erstellten Bildes dient die schon erwähnte Kirche (Listing 1). Das Programm ist so gestaltet, daß es auch als Unterprogramm in Ihren eigenen Programmen verwendet werden und nicht nur Kirchen zeichnen kann. Geben Sie zuerst die Zeilen ab 60000 bis 60200 ein und speichern Sie diese (SAVE "CHARGRAF"). Mit dem Befehl **MERGE "CHARGRAF"** können Sie nun die Routine in Ihr eigenes Programm einbinden und zusammen mit diesem speichern, vorausgesetzt, die Zeilennummern sind nicht schon vergeben. In diesem Fall kann der von der Routine belegte Zeilenbereich mit Hilfe des RENUM-Befehls (s. Basic-Handbuch) leicht geändert werden.

Die Grafik-Routine kann dann einfach mit GOSUB 60000 (oder der mit RENUM neu vergebenen Zeilennummer) aufgerufen werden. Vorher muß jedoch, wenn sich in Ihrem Programm noch andere Daten in Form von DATA-Zeilen befinden oder dieses mehrere Bilder enthält, der Datenzeiger von Basic mit dem RESTORE-Befehl auf die richtigen DATA-Zeilen gestellt werden.

Funktion

Die Routine ist in der Lage, von einem Startpunkt x,y (Zeile, Spalte des Bildschirms) in eine von acht möglichen Richtungen eine gewisse Anzahl gleicher Zeichen zu "plotten". Daraus ergeben sich für jedes darzustellendes Bilddetail fünf notwendige Daten (Parameter), die von der Routine mit READ aus den DATA-Zeilen gelesen werden.

Der erste Parameter bestimmt die Zeile (0 – 31), der zweite die Spalte (0 – 89), ab der gezeichnet werden soll. Der nächste Parameter bestimmt eine der acht möglichen Richtungen, in die gezeichnet werden soll. Die Werte der Richtungen finden Sie in Bild 2.

Als nächstes wird das Zeichen bestimmt, mit welchem geplottet werden soll. Dazu wird der dezimale Wert des Zeichens angegeben, den Sie unserer Tabelle und der Zeichensatztabelle des Benutzer-Handbuches entnehmen können. Für den Buchstaben "A" wäre der Wert 65 anzugeben.

Zum Schluß folgt noch die Anzahl der darzustellenden Zeichen ab der Startposition in die angegebene Richtung. Der Wert 1 würde die Ausgabe eines Zeichens bewirken, mit 5 würden entsprechend fünf Zeichen in die angegebene Richtung ausgegeben.

Damit die Routine ihre Arbeit auch mal wieder beendet und die Kontrolle zum aufrufenden Programm wieder zurückkehrt, ist als letzter Wert eines zu zeichnenden Bildes für die Zeilenposition der Wert 99 anzugeben (s. Zeile 60070). Dieser wurde gewählt, da er außerhalb der zulässigen Bildschirmkoordinaten liegt und in einer korrekten Grafik nicht möglich ist. Beachten Sie bitte auch, daß eine Über-/Unterschreitung der zulässigen Bildkoordinaten durch eine zu große Anzahl zu zeichnender Zeichen in eine Richtung von der Routine nicht überprüft werden und es daher zu ungewollten Effekten kommen kann.

Anregungen...

Da sich die Funktion der Routine nur auf das Notwendigste beschränkt, wäre es interessant, sie noch um weitere "Features" zu erweitern. Z.B. könnte sie noch um die Möglichkeit der inversen und unterstrichenen Darstellung oder um weitere Logo-ähnliche Funktionen erweitert werden... (Jürgen Keller/MC)

Listing 1: KIRCHE.BAS

```
20 ' Demografik: Kirche
100 RESTORE 500:GOSUB 60000
110 END
490 '
500 'Datazeile: Zeile, Spalte, Richtung, Zeichen,
Anzahl
510 DATA 8,12,6,134,1,9,12,6,133,18,28,11,4,131,1,
28,12,4,61,7,28,19,4,137,1,28,20,2,133,18,10,20,2,
140,1,9,20,8,138,7,10,10,3,47,5,10,22,1,211,5,3,16
,6,167,1:REM turm
515 DATA 13,17,1,1,1
520 DATA 28,20,4,138,45,28,65,4,137,1,28,66,2,133,
6,22,66,2,140,1,21,66,8,138,45,18,20,4,95,43,18,63
,5,211,4:REM Kirchenschiff
530 DATA 4,31,6,35,5,4,35,7,35,2,6,31,5,35,3,4,36,
6,35,1,6,36,6,35,3,6,38,6,35,3,7,41,8,35,2,6,42,6,
35,3,7,42,4,35,2,9,42,4,35,2,4,46,6,35,5,6,47,6,35
,1,6,48,6,35,3,6,50,6,35,3,6,51,6,35,1,7,51,6,34,1
,8,51,6,35,1,6,52,6,35,1,8,52,6,35,1,99:REM schrif
t
```

```
60080 '          G R A F I K          by JK(1986)
60040 esc$=CHR$(27):home$=esc$+"H":cls$=esc$+"E"+h
ome$
60050 ch$=esc$+"A":ct$=esc$+"B":cr$=esc$+"C":cl$=e
sc$+"D":PRINT cls$
60060 'X- und Y-Koordinate:
60070 READ x:IF x=99 THEN RETURN
60080 READ y:PRINT esc$+"Y";CHR$(32+x);CHR$(32+y);
: 'positionieren
60090 'c Zeichen b in Richtung a zeichnen:
60100 READ a:READ b:READ c:FOR i=1 TO c:ON a GOSUB
 60130,60140,60150,60160,60170,60180,60190,60200 '
Stift bewegen
60110 IF b<32 THEN PRINT CHR$(27);CHR$(b);CHR$(8);
 ELSE PRINT CHR$(b);CHR$(8); 'Zeichen setzen
60120 NEXT:PRINT CHR$(13):GOTO 60070 'nächster Str
ich
60130 PRINT ch$;CHR$(8);:RETURN:REM links schräg h
och
60140 PRINT ch$;:RETURN:REM hoch
60150 PRINT ch$;cr$;:RETURN:REM rechts schräg hoch
60160 PRINT cr$;:RETURN:REM rechts
60170 PRINT cr$;ct$;:RETURN:REM rechts schräg runt
er
60180 PRINT ct$;:RETURN:REM runter
60190 PRINT ct$;CHR$(8);:RETURN:REM links schräg r
unter
60200 PRINT cl$;:RETURN:REM links
```

Tabelle: Grafik-Zeichen

dicker Strich:		
━	CHR$(154)	waagerecht voll
╸	CHR$(152)	waagerecht halb
┃	CHR$(149)	senkrecht voll
╹	CHR$(145)	senkrecht halb oben
╻	CHR$(148)	senkrecht halb unten
┗	CHR$(147)	Ecke links unten
┏	CHR$(150)	Ecke links oben
┛	CHR$(153)	Ecke rechts unten
┓	CHR$(156)	Ecke rechts oben
┣	CHR$(151)	Abzweig rechts
┫	CHR$(157)	Abzweig links
┻	CHR$(155)	Abzweig oben
┳	CHR$(158)	Abzweig unten
╋	CHR$(159)	Kreuzung

dünner Strich:		
-	CHR$(45)	kl. Strich waagerecht mitte
_	CHR$(95)	gr. Strich waagerecht unten
ǀ	CHR$(243)	Strich senkrecht
/	CHR$(47)	Schrägstrich rechts
\	CHR$(211)	Schrägstrich links
(	CHR$(60)	Knick links
)	CHR$(62)	Knick rechts
X	CHR$(88)	Doppelknick

Doppelstrich:		
═	CHR$(138)	waagerecht
║	CHR$(133)	senkrecht
╚	CHR$(131)	Ecke links unten
╔	CHR$(134)	Ecke links oben
╝	CHR$(137)	Ecke rechts unten
╗	CHR$(140)	Ecke rechts oben
╠	CHR$(135)	Abzweig rechts doppelt
╣	CHR$(141)	Abzweig links doppelt
╟	CHR$(130)	Abzweig rechts einfach
╢	CHR$(136)	Abzweig links einfach
╩	CHR$(139)	Abzweig oben doppelt
╦	CHR$(142)	Abzweig unten doppelt
╧	CHR$(129)	Abzweig oben einfach
╤	CHR$(132)	Abzweig unten einfach
╬	CHR$(143)	Kreuzung

Sonderzeichen:		
≡	CHR$(255)	Dreifachstrich waagerecht
=	CHR$(61)	Doppelstrich waagerecht
"	CHR$(34)	Doppelstrich senkrecht oben
#	CHR$(35)	Doppelkreuz
¨	CHR$(178)	Doppelpunkt oben liegend
:	CHR$(58)	Doppelpunkt stehend
·	CHR$(144)	kleiner Punkt mitte
•	CHR$(128)	großer Punkt mitte
«	CHR$(171)	Doppelpfeil links
»	CHR$(172)	Doppelpfeil rechts
*	CHR$(42)	Stern
+	CHR$(43)	kleines Kreuz
°	CHR$(162)	kleiner Kreis oben
✓	CHR$(174)	Hacken
Y	CHR$(89)	Sektglas
¥	CHR$(189)	Pokal
†	CHR$(167)	Kreuz
○	CHR$(187)	Kreis leer
●	CHR$(188)	Kreis voll
⇐	CHR$(253)	Pfeil doppelt links
⇒	CHR$(252)	Pfeil doppelt rechts

Iscape-Sonderzeichen:		
⊙	CHR$(27);CHR$(1)	Kreis mit Punkt
Γ	CHR$(27);CHR$(2)	Galgen
Δ	CHR$(27);CHR$(3)	Dreieck
⊕	CHR$(27);CHR$(4)	Kreis mit Kreuz
×	CHR$(27);CHR$(5)	Kreuz
÷	CHR$(27);CHR$(6)	Doppelpunkt mit Strich
∴	CHR$(27);CHR$(7)	Dreieck als Punkte
←	CHR$(27);CHR$(11)	Pfeil links
→	CHR$(27);CHR$(12)	Pfeil rechts
↔	CHR$(27);CHR$(14)	Pfeil links/rechts

FONTLIB

- Zeichensatzverwaltung für DR. Draw auf dem Joyce

Die acht Zeichensätze, mit denen man seine mit DR. Draw erstellten Zeichnungen beschriften kann, sind eine feine Sache. Nicht ganz so schön ist die Tatsache, daß immer nur drei Schriftarten auf einmal verwendet werden können. Und schlimm wird es erst, wenn man mal die Schriftarten wechseln will. Kreuz und quer, hin und her kopieren, Files umbenennen, dann einen Merkzettel an die Pinwand "FONTA ist GOTITA"... oder auch nicht ? Fangen wir also nochmal von vorne an, usw...
Dies muß aber nicht sein. Das Programm FONTLIB, geschrieben in Mallard-Basic für den JOYCE, nimmt dem Anwender diesen ganzen "Verwaltungskram" ab. Sobald das Listing eingetippt und die Font-Disk (dt.: Zeichensatz-Diskette) erstellt ist, läuft alles (fast) wie von selbst.

Die Vorarbeiten:

Zunächst müssen alle acht Zeichensatz-Dateien von der DR. Draw-Diskette auf einer neuen Diskette versammelt werden. Bei einer Länge von 21 kByte pro Zeichensatz macht das genau 168 kByte. Bleiben also auf einer 172 kByte-Diskette noch 4 kByte für die Verwaltung der Zeichensätze übrig. Das hört sich zwar knapp an, ist aber kein Hindernis, wie sich später zeigen wird.
Aber fangen wir mit dem Kopieren an:
Man nehme eine formatierte Leerdiskette und die Original-DR. Draw-Diskette (letzteres ist wichtig, da es sonst beim ersten Programmlauf zum Chaos kommen kann: die angenommene Zeichensatzbelegung könnte ja schon von Ihnen auf Ihrer Arbeitskopie geändert worden sein). Zunächst werden die unbenutzten Zeichensätze, erkennbar an dem Dateityp ".8" von Seite 1 des Originals auf die neue Diskette kopiert (s. auch Startdiskette anders erstellt, Ausgabe 6/86 und Dr. Draw-Dokumentation). Auf die gleiche Weise gelangen die voreingestellten Zeichensätze (Default-Fonts) von Seite 2 des Originals (erkennbar an dem Dateityp ".BIN") auf unsere Zeichensatz-Diskette. Damit wäre die gröbste Arbeit schon mal geschafft!

...und nun zum Programm:

Seine Aufgaben sind folgende:
1. Die aktuelle Belegung der Default-Fonts einlesen.
2. Diese momentan aktiven Fonts in ihren ursprünglichen Namen umbenennen.
3. Vom Benutzer die neuen gewünschten Fonts erfragen.
4. Die Dateien entsprechend umbenennen, so daß sie für DR. Draw die Default-Fonts werden.
5. Die neue Belegung speichern.

Und das war's auch schon: Der Benutzer wählt drei Zeichensätze aus, das Programm übernimmt die Verwaltung und der Benutzer braucht nur noch die drei Dateien mit den neuen aktiven Zeichensätzen auf seine Arbeitsdiskette kopieren.
Das Gedächtnis von FONTLIB ist die Datei BELEGUNG-.AKT. Sie enthält die internen Nummern der drei belegten Default-Fonts. Aus dieser Nummer ergibt sich der ursprüngliche Name usw. Eine genauere Beschreibung möchte ich mir hier sparen, denn ich habe (entgegen sonstigen Gewohnheiten ???) ein recht gut dokumentiertes Listing erstellt.
Der erfahrene Programmierer erkennt die Funktionsweise des Programmes sicher schnell, Unerfahrene sollten das Funktionieren des Programms als "mystische Manifestation der Allmacht der Technik" (oder so ähnlich) interpretieren. Leider hat die Dokumentation einen Nachteil: Das Programm ist zu lang. Wir haben noch 4 kByte frei, davon geht 1 kByte für das Gedächtnis drauf. Länger als 3 kByte darf die Verwaltung also nicht sein.

Daher also einige Zugeständnisse an Platzmangel und Tippfaulheit:

Die REM- und Leerzeilen werden nicht angesprungen und können somit weggelassen werden. Zweitens ist das Unterprogramm "Kaltstart" nur beim ersten Durchlauf nötig. Man kann also das Programm auf eine andere Diskette speichern, von dort aus laden und vor dem Start bzw. spätestens bei der entsprechenden Aufforderung die Zeichensatzdiskette einlegen, damit die Gedächtnisdatei an die richtige Stelle kommt. Wichtig ist, daß Sie in dieser Phase einen kompletten Programmdurchlauf machen, d.h. (irgend)eine vom Programm angebotene Wahl treffen und nicht vor dem offiziellen Ende aufhören, z.B. durch Drücken der STOP-Taste oder Ausschalten des Rechners.

Als nächstes löschen Sie das Unterprogramm und seinen Aufruf (siehe Listingkommentar) und, soweit nicht schon geschehen, die Kommentare und speichern das neue Programm mit auf die Zeichensatzdiskette (Tippfaule erhalten mit der DATABOX gleich beide Versionen...). Die neue Programmdatei ist dann nur noch 3 kByte lang und paßt so gerade noch auf die Zeichensatzdiskette.
Wenn dies alles geschafft ist, wird der Schriftwechsel zum Kinderspiel:

- CP/M booten (Rechner einschalten und Systemdiskette einlegen).
- PIP.COM in die RAM-Disk kopieren (**pip m:=pip.com**)
- Mallard-Basic laden (**basic**)
- Zeichensatzdiskette einlegen und FONTLIB laden (**load "fontlib"** oder **run "fontlib"**)
- Nach dem Start gewünschte Zeichensätze wählen
- Nach Arbeitsende die drei neuen Font-Dateien mit PIP zunächst in die RAM-Disk (**pip m:=*.bin**) und nach einlegen Ihrer Arbeitsdiskette von dort auf selbige kopieren (**pip a:=m:*.bin**)

Falls einmal während der Arbeit der Strom ausfallen sollte, oder wenn aus irgendeinem Grund einmal alle acht Dateien mit ihren unbenutzten Namen auf der Diskette stehen sollten, sich also keine .BIN-Datei finden läßt, so ist dies keine ausweglose Situation.
In diesem Fall laden Sie FONTLIB und setzen im Listing nach dem Programmteil Deklarationen ein GOTO <***MENU***>. Damit überspringen Sie den Teil, der die Dateien in den Ursprungszustand zurückversetzt und können wie gewohnt weiterarbeiten. Allerdings dürfen Sie vorher keine Namen verändern!!!

Noch ein Tip: Falls Sie DR. Draw auf der RAM-Disk installiert haben, brauchen Sie auf Ihrer Arbeitsdiskette nur noch die drei aktiven Font-Dateien. Das bedeutet für Sie: mehr als 100 kByte Platz für Bilder und Sie können die Aufforderung zum Disketten-Wechsel einfach ignorieren.

(Michael Anton/MC)

```
10 REM *******************************************
20 REM ***   FONTLIB  -  DR Draw Font Library  ***
30 REM ***        1986 by Michael Anton        ***
40 REM *******************************************
50 :
60 :
70 REM ---------------------------INITIALISIERUNG
-----------------------------------------------
80 :
90 :
100 REM *** Deklarationen ***
110 :
120 DIM bez$(8),font$(3),klar$(8),bel(3)
130 :
140 :
150 REM *** Filenamen lesen ***
160 :
170 DATA "SIMROM.8","DUPROM.8","COMROM.8","INDROM.
8","INDITA.8","SIMSCR.8"
180 DATA "COMSCR.8","GOTITA.8","FONTA.BIN","FONTB.
BIN","FONTC.BIN"
190 :
200 FOR lauf=1 TO 8 : READ bez$(lauf) : NEXT lauf
210 FOR lauf=1 TO 3 : READ font$(lauf) : NEXT lauf
220 :
230 REM *** Klartextbezeichnung lesen ***
240 :
250 DATA SIMPLEX ROMAN,DUPLEX ROMAN,COMPLEX ROMAN,
INDEXICAL ROMAN
260 DATA INDEXICAL ITALIC,SIMPLEX SCRIPT,COMPLEX S
CRIPT,GOTHIC ITALIC
270 :
280 FOR lauf=1 TO 8 : READ klar$(lauf) : NEXT lauf
290 :
300 REM *** Steuerzeichen ***
310 :
320 cls$=CHR$(27)+"E"+CHR$(27)+"H"
330 beep$=CHR$(7)
340 revon$=CHR$(27)+"p"
350 revoff$=CHR$(27)+"q"
360 :
370 REM *** Filenamen ***
380 :
390 bf$="BELEGUNG.AKT"
400 :
410 :
420 :
430 REM ------------------------------HAUPTPROGRAMM
-----------------------------------------------
440 :
450 :
460 REM *** Start ***
470 :
480 PRINT cls$,beep$ : FOR lauf=1 TO 13 : PRINT :
NEXT lauf
490 INPUT "Bitte Diskette mit Font Files einlegen
und <RETURN> drücken: ",dummy$
```

```
500 :
510 RESET
520 :
530 IF FIND$(bf$)="" THEN GOSUB 1520 : REM *** ->
Kaltstart ***
540 OPTION RUN
550 :
560 REM *** Belegung einlesen ***
570 :
580 OPEN "I",1,bf$
590  FOR lauf=1 TO 3 : INPUT #1,bel(lauf) : NEXT l
auf
600 CLOSE 1
610 :
620 REM *** Files umbenennen ***
630 :
640 FOR lauf=1 TO 3
650     NAME font$(lauf) AS bez$(bel(lauf))
660 NEXT lauf
670 :
680 KILL bf$
690 :
700 :
710 REM *** Auswahl ***
720 :
730 PRINT beep$;cls$;revon$;
740 PRINT "+----------------------------+"
750 PRINT "! ** DR Draw Font Library ** !"
760 PRINT "! **    1986 by M.Anton    ** !"
770 PRINT "+----------------------------+";revoff$
780 PRINT : PRINT : PRINT
790 PRINT "Folgende Schrifttypen stehen zur Auswah
l:" : PRINT
800 :
810 FOR lauf=1 TO 8
820     PRINT TAB(25);revon$;lauf;revoff$;" = ";kla
r$(lauf)
830 NEXT lauf
840 :
850 PRINT : PRINT : PRINT revon$;
860 PRINT "Bitte wählen Sie durch Eingabe der ents
prechenden Ziffer, welche Schriftart"
870 PRINT "DR Draw Ihnen unter den folgenden Menup
unkten anbieten soll:";revoff$;beep$
880 :
890 INPUT "Als 'SIMPLEX': ",bel(1)
900 IF bel(1)<1 OR bel(1)>8 THEN GOTO 890
910 INPUT "Als 'COMPLEX': ",bel(2)
920 IF bel(2)<1 OR bel(2)>8 THEN GOTO 910
930 INPUT "Als 'ITALIC' : ",bel(3)
940 IF bel(3)<1 OR bel(3)>8 THEN GOTO 930
950 :
960 IF bel(1)<>bel(2) AND bel(2)<>bel(3) AND bel(3
)<>bel(1) THEN GOTO 1060
970 :
980 PRINT beep$;revon$;" MEHRFACHZUWEISUNG UNZULÄ
SSIG !!!"
990 PRINT "  Bitte Taste drücken...";revoff$;beep$
1000 IF INKEY$="" THEN GOTO 1000
1010 GOTO 730
1020 :
1030 :
1040 REM *** Check ***
1050 :
1060 PRINT cls$
1070 PRINT : PRINT : PRINT : PRINT "    Die neue B
elegung ist: " : PRINT : PRINT
1080 PRINT TAB(20);"Menupunkt","Schriftart"
1090 PRINT TAB(20);"------------------------------
----"
1100 PRINT TAB(20);" SIMPLEX:",klar$(bel(1))
1110 PRINT TAB(20);" COMPLEX:",klar$(bel(2))
1120 PRINT TAB(20);" ITALIC :",klar$(bel(3))
1130 :
1140 PRINT : PRINT : PRINT revon$;"Ist diese Beleg
ung in Ordnung???"
1150 PRINT "Für neue Belegung <N> eingeben, sonst
nur <RETURN>...";revoff$;
1160 INPUT dummy$
1170 IF dummy$="N" OR DUMMY$="n" THEN GOTO 730
1180 :
1190 :
1200 REM *** Files umbenennen ***
1210 :
1220 FOR lauf=1 TO 3
1230     NAME bez$(bel(lauf)) AS font$(lauf)
1240 NEXT lauf
1250 :
1260 REM *** neue Belegung speichern ***
1270 :
1280 OPEN "O",1,bf$
1290  FOR lauf=1 TO 3 : PRINT #1,bel(lauf) : NEXT
lauf
1300 CLOSE 1
1310 :
1320 :
1330 REM ---------------------------------PROGRAMME
NDE-----------------------------------
1340 :
1350 :
1360 OPTION STOP
1370 PRINT : PRINT beep$;"Bitte kopieren Sie jetzt
 die neuen Font Files mit";revon$
1380 PRINT "M:PIP M:=*.BIN /Diskwechsel/ M:PIP A:=
M:*.BINÄWü";revoff$
1390 PRINT "auf Ihre DR Draw Arbeitsdiskette. (Son
st war alles umsonst...)"
1400 :
1410 END : REM ----> evtl. durch 'SYSTEM' ersetzen
1420 '                  um Basic automatisch zu verl
assen
1430 :
1440 :
1450 :
1460 REM -----------------------------UNTERPROGRAMM
 KALTSTART---------------------------
1470 :
1480 :
1490 :
1500 REM *** Default-Belegung erzeugen ***
1510 :
1520 OPEN "O",1,bf$
1530 PRINT #1,1 : REM --> FONTA.BIN=SIMROM.8
1540 PRINT #1,3 : REM --> FONTB.BIN=COMROM.8
1550 PRINT #1,7 : REM --> FONTC.BIN=COMSCR.8
1560 CLOSE 1
1570 RETURN
1580 :
1590 REM Es wird hier vorausgesetzt, daß die Font
Files ('FONTX.BIN') und die anderen
1600 REM Files in ihrer Benennung denen der Origin
al DR Draw Diskette entsprechen
1610 REM und auf der FontDisk vollständig vorhande
n sind.
1620 :
1630 REM Eine Vollständigkeitsprüfung wäre zwar mö
glich, ist aber für den einmaligen
1640 REM Gebrauch vollkommen überflüssig. Hier ist
 das Können des Anwenders gefragt...
1650 REM Im übrigen kann dieses Unterprogramm nach
 der ersten Anwendung entfernt
1660 REM werden, es muß allerdings auch sein Aufru
f wegfallen. (Zeile 460-520)
1670 REM Auch müssen die REM-Zeilen entfernt werde
n, damit das Programm noch auf die
1680 REM Diskette mit den 8 Font Files passt. PIP
sollte in der RAM-Disk stehen.
1690 :
1700 REM Die REM- & Leerzeilen werden an keiner St
elle angesprungen und können
1710 REM somit entfallen.
1720 :
1730 REM                                    Viel V
ergnügen...
```

Praktische Textverarbeitung mit

JOYCE

Schreib-Praxis Paket*

Buch & Diskettenpaket zur praxisnahen Anwendung von LocoScript

Ein Novum unter den Computerbüchern!

Bunt gemischtes Anwender-Paket für den Joyce-Texter: **Buch + Diskette** mit Wort-Experimenten, Tips und Tricks, einem Layout-Archiv, dem LocoScript-Software-Training, literarischen Text-Beispielen, »historischen Simulationen, Insider-Plaudereien und feuilletonistischen Abwegen«, theoretischen Notizen und vieles mehr.

Ein Wegweiser durch die Welt der Textverarbeitung.

Abwechslungsreicher Lesestoff für Neulinge und Fortgeschrittene, der über die Grenzen des Computerschreibtischs hinausführt.

Auf Diskette:

1. Der **»LocoScript-Zettelkasten«** bietet direkten Zugriff auf zahlreiche LocoScript-Standard-Schablonen in über 50 Dateien. Dazu: Mustertexte, Editierübungen, Schriftbeispiele, Serienbriefe, Spaltendruck, Tabellen, Telefonregister, typographische Experimente, Text-Kostproben, und vieles mehr.
2. **WordStar-Tastendefinitionsdatei** mit Referenzkarte, WS-Stapeldatei für Startdiskette (beides individuell anzupassen).
3. **LOGO- und BASIC-Programme** zur Erstellung computergenerierter Sätze und Gedichte.



*Buch + Diskette für LocoScript und WordStar-Autoren von DMV

Ca. 200 Seiten, Einband: Leinen-Hardcover
Buch und 3"-Diskette zum Preis von 89,– DM

Lieferbar ab Mitte September 1986.
Zu beziehen über den Computerhandel und den guten Fachbuchhandel oder direkt beim Verlag. Händleranfragen erwünscht.

DMV Verlag, Fuldaer Str. 6, Postfach 250, 3440 Eschwege



Serienbriefe mit JOYCE ...und noch einiges mehr.

Die Schwäche von LocoScript, das Erstellen von Serienbriefen mit Hilfe von Adressdateien, der Betrieb von Fremddruckern ohne den Verlust der Layout-Möglichkeiten von LocoScript und die Fähigkeit, Texte mittels DFÜ zu übertragen, wurde schon mehrmals zu beheben versucht. Die, meiner Meinung, bis jetzt gelungendste Lösung dieser Problematik stellt ein Programmpaket der Star-Division dar, bestehend aus Star-Mail und Datei-Star. Mit diesen Programmen wurden oben genannte, fehlende Funktionen sehr gut realisiert.

Es sei gleich vorweg genommen:
Ein Eingehen auf die Programme bis ins letzte Detail ist hier aus Platzgründen nicht möglich. Das Hauptaugenmerk soll auf Star-Mail liegen, welches "die Schnittstelle" zwischen LocoScript und anderer Hard- und Software darstellt. Der Datei-Star ist, wie der Name vermuten läßt, ein Datenverwaltungsprogramm, mit dem die für Serienbriefe nötigen Daten erfaßt und verwaltet werden. Er kann aber auch als eigenständiges Programm benutzt werden.
Die Dokumentation beider Programme stellt alle Arbeitsabläufe übersichtlich dar und dürfte auch für weniger erfahrene JOYCE-Besitzer verständlich sein. Bei Star- Mail steht sogar ein Übungsteil zur Verfügung, was leider (noch?) nicht bei jedem Produkt selbstverständlich ist.

Star-Mail

Mit Star-Mail steht ein umfangreiches Programm zur Verfügung, das neben der Serienbrieferstellung auch Mailing, Verwaltung von Steuerzeichen, Druckeranpassung usw. ermöglicht. Alle Funktionen des Programmes werden über Menues ausgewählt. Lediglich die in den zu druckenden Text bei der Erstellung mit LocoScript einzugebenden Star-Mail-Befehle (s.u.) müssen "gelernt" werden.

Angenehm fällt die in Star-Mail integrierte Diskettenverwaltung mit Funktionen wie das Ausdrucken des Inhaltverzeichnisses, Löschen und Umbenennen von Dateien auf. So braucht man z.B. nicht das Programm wegen Platzmangel auf der Diskette verlassen, nicht mehr benötigte Dateien löschen und dann das Programm wieder starten. Diese Funktionen sind, aus mir unerfindlichen Gründen, bei vielen anderen JOYCE-Programmen oft nicht implementiert.

Der Text für Serienbriefe wird wie gewohnt mit LocoScript erstellt, wobei allerdings ein paar Gestaltungsmöglichkeiten LocoScripts von Star-Mail nicht unterstützt werden. Dies betrifft u.a. Kopf-, Fußzeilen sowie Proportionalschrift im Blocksatz. Ebenso kann nicht der gesamte, von LocoScript bereitgestellte, Zeichensatz verwendet werden. Dieser kann von Fremd-Druckern oder anderen Programmen im allgemeinen sowieso nicht verarbeitet werden. Für den Verwendungszweck von Star-Mail stellen diese Einschränkungen also keine großen Hindernisse dar. Als "Wiedergutmachung" bietet das Programm dafür einige Funktionen, die man in LocoScript vergeblich sucht.
An den Textstellen, an denen Daten aus einer Datei-Star-Datei während des Druckens eingefügt werden sollen, werden die einzelnen Daten-Felder durch Platzhalter definiert. Dabei ist die Reihenfolge der Felder gleichgültig und auch eine mehrfache Verwendung möglich. Zur Steuerung des Datenzugriffs auf eine Datei-Star-Datei sowie vieler anderer, nützlicher Möglichkeiten stehen weitere Kommandos zur Verfügung. So kann der Druck zwecks Typenradwechsel bei Verwendung eines solchen Druckers angehalten werden. "Speicherplätze" (ähnl. Variablen in Programmiersprachen) können mit Daten (Texte oder Zahlen) belegt, rechnerisch verknüpft und an beliebiger Stelle im Text gedruckt werden. Ebenso können individuelle Eingaben während des Drucks vorgenommen werden. Der den eingesetzten Daten folgende Text wird wieder entsprechend des Layouts formatiert, man braucht sich also keine Gedanken, um die Längen der eingesetzten Daten zu machen. Lücken oder abgeschnittene Datenschnipsel entstehen nicht.

Damit bei Verwendung von Fremddruckern deren Möglichkeiten voll genutzt werden können, bietet Star-Mail eine umfangreiche und leicht zu bedienende Anpassungsmöglichkeit selbiger. Dies umfaßt die Steuerzeichen sowie eine Konvertierungstabelle der ASCII-Codes in die spezifischen Drucker-Codes. Diese Parameter-Liste kann für verschiedene Drucker erstellt, gespeichert und bei Bedarf geladen werden.
Ist der Text erstellt und der Drucker eingestellt, so wird von Star-Mail die LocoScript-Datei konvertiert (Erstellung einer ASCII-Datei ist also wie bei anderen Lösungen nicht notwendig bzw. erlaubt!) und, wenn nötig, aus einer Star-Datei die gewünschten Daten ausgelesen. Beides wird temporär in Dateien der RAM-Disk abgelegt, um schließlich wahlweise auf JOYCE- oder Fremd-Drucker, seriellen Schnittstelle oder in eine Datei ausgegeben zu werden.

Wertung:

Mit diesem Programmpaket werden die Einsatzmöglichkeiten des JOYCE wesentlich erweitert und ein effektives Arbeiten, z.B. bezüglich der Geschäftskorrespondenz, erst möglich.
Hersteller: Star-Division
Preis: Star-Mail 98,- DM
Datei-Star 98,- DM
zusammen 189,- DM

(MC)



Top-Programm des Monats

Paint Master

für 464-664-6128

Paint Master ist ein Malprogramm für den Schneider Computer CPC 464/664/6128, mit oder ohne Diskettenlaufwerk. Es besteht im wesentlichen aus drei Teilen :

1. Paint: ist das Ladeprogramm.

2. Code: ist eine Binärdatei, die die RSX-Befehle für das Hauptprogramm enthält.

3.Master: ist das eigentliche Hauptprogramm.

Nach dem Laden ist Mode 1 selektiert und in der Mitte des Bildschirms sind zwei Kreuze abgebildet: Ein kleines mit dem Namen ORIGIN und ein großes Kreuz mit dem Namen TARGET; letzteres läßt sich mit Hilfe der Cursortasten oder eines Joysticks bewegen.

Wenn Sie die Space-Taste drücken, erscheint der Helpscreen auf dem Bildschirm. Hier finden Sie alle Malhilfen mit den dazugehörigen Tastenbelegungen. Die Malhilfen unterteilen sich in zwei Gruppen: Einmal die mit direkter Wirkung (d.h. sofort nach dem Drücken der entsprechenden Taste) und zum anderen mit indirekter Wirkung, wobei die Funktion erst mit Drücken der Copycursortaste oder des Feuerknopfes aktiviert wird.

Beschreibung der Funktionen mit Direktwirkung:

ZOOM:

Der gesamte Grafikbildschirm ist in 64 gleichgroße Bereiche unterteilt und jener, in dem sich TARGET befindet, wird achtfach vergrößert dargestellt. Ein blinkender Cursor markiert die ständige Position. Punkte werden in der aktuellen Penfarbe mit der Copy-Taste (bzw. Feuerknopf) gesetzt oder mit der DEL-Taste gelöscht. Nach zweimaligem Drücken der großen ENTER-Taste wird das Bild wieder verkleinert und mit den Korrekturen an die alte Stelle geblendet.

SWAP:

Vertauscht die Bereiche, in denen sich Origin und Target befinden sowie deren Koordinaten

FIND:

Umrahmt den Teilbereich, in dem sich Target befindet.

TEXT:

Alle 256 Symbole des Zeichensatzes können mittels Steuerung frei ausgewählt werden. Anschließend kann die so ausgewählte Zeichenkette in Originalgröße mittels Copy-Taste im Ursprung von Target ausgegeben werden.

SYMBOL:

Ermöglicht die Definition eigener Grafikzeichen. Nachdem das gewünschte Symbol mit Hilfe der Cursortasten ausgewählt wurde, beginnt der Entwurf des Zeichens. Hierzu ist der Cursor in einer 9*8-Matrix abgebildet. Jetzt gibt man einen 64 Zeichen langen String ein, der aus Nullen und Einsen besteht und schließt mit ENTER ab.

Das Aussehen des Zeichens wird durch die Zahlen (0=paper/1=pen) bestimmt.

PEN:

Der alte Farbstift wird angezeigt, einen neuen kann man in den Grenzen des Mode wählen.

INK:

Anzeige der INK- und Farbtabelle. Änderung durch Eingabe.

BORDER:

Rahmenfarbe und Farbtabelle werden ausgegeben und können geändert werden.

MODE:

Ändern des Screen-Mode und Löschen der Grafik.

WRITE:

Schreibmodus der Grafik VDU kann geändert werden. Defaultwert ist 0 (Normal). Die Modi 1 - 3 entsprechen den logischen Verknüpfungen XOR, AND und OR.

STEP:

Die Schrittweite des Cursors kann im Rahmen von 1 - 80 Pixel geändert werden. Standardwert ist 4.

SAVE:

Nach der Eingabe des Dateinamens, der aus Gründen der

Diskettenkompatibilität nicht länger als acht und minimal ein Zeichen lang sein muß, wird abgefragt, ob man das komplette Bild oder nur einen Teilbereich abspeichern will. Ein Teilbereich wird durch die Teilbereiche begrenzt, die von dem Rechteck zwischen Origin und Target berührt werden.
LOAD:
Lädt eine Bilddatei
MERGE:
Lädt ein Teilbild wieder an seine ursprüngliche Position, ohne den Rest des Gesamtbildes zu beeinflussen.
MOVE:
Verschiebt Origin nach Target
SETUP:
Ist die jeweilige indirekte Funktion, die vorher angewählt wurde und erst mit Drücken der Copy-Taste gestartet werden kann. Die einzelnen Möglichkeiten sind: PLOT, DRAW, REC, BOX, CUBE, CIRCLE, SPOT, FILL, SPRAY, PRINT, LARGE, CLG, PCLS, INVERT und COPY.
ERASE:
Ist das gleiche wie Setup, nur das hier mit PEN 0 gearbeitet wird.
HELP:
Aktiviert die Helpscreen
EXIT:
Zweimaliges Drücken der kleinen ENTER-Taste ermöglicht ein Verlassen des Programms.

Einzelbeschreibung der Funktionen mit indirekter Wirkung:
PLOT:
Setzt einen Punkt in das Zentrum von Target
DRAW:
Verbindet Origin und Target mit einer Linie
REC:
Zeichnet ein Rechteck mit den Eckkoordinaten von Origin und Target
BOX:
Wie REC nur ausgefüllt
CUBE:
Zeichnet einen Quader in 3D-Darstellung. Origin ist hierbei ein Eckpunkt der Frontfläche.

Der erste Standort von Target gibt die Richtung der schräg verlaufenden Seitenlinien ausgehend von Origin an, und der zweite Standort von Target definiert den zweiten Eckpunkt der Frontfläche.
CIRCLE:
Zeichnet einen Kreis bzw. eine Ellipse mit dem Zentrum Origin und dem Radius, der sich aus dem horizontalen und vertikalen Abstand von Target zu Origin ergibt.
SPOT:
Wie CIRCLE nur ausgefüllt
FILL:
Füllen einer geschlossenen Fläche in Penfarbe ausgehend von Target. Bei komplizierten Flächen mehrmals füllen.
SPRAY:
Die Fläche zwischen Origin und Target wird mit zufälligen Plotpunkten gesprenkelt.
LARGE:
Gibt die Zeichen, die mit TEXT eingegeben wurden, in vergrößerter Darstellung aus. Origin bildet den unteren rechten Eckpunkt des ersten Zeichens an, Target den linken oberen Eckpunkt. Andere Stellungen von Origin und Target bringen Spiegelungseffekte.
CLG:
Das Rechteck zwischen Origin und Target wird gelöscht.
PCLS:
Löscht den Teilbereich, in dem sich Target befindet.
INVERT:
Invertiert den Teilbereich, in dem sich Target befindet.
COPY:
Kopiert den Inhalt des Teilbereichs, in dem sich Origin befindet, nach dem Teilbereich von Target.
Beschreibung der RSX-Befehle:
GPEN,n:
Setzt den Grafikfarbstift
FILL:
Füllt eine umschlossene Fläche oder bis an die Grenzen des Grafikwindows
KEEP,n:
Kopiert den Teilbereich n in einen 250 Bytes großen Buffer ab Adresse 32768
SHOW,n:
Lädt Teilbereich n mit dem Inhalt des Buffers.
SWAP,n:
Vertauscht Teilbereich n mit Inhalt des Buffers
ZOOM:
Vergrößert den Inhalt des Buffers um das Achtfache
REDUCE:
Verkleinert den Bildschirm und legt den Inhalt im Buffer ab
PCLS,n:
Löscht Teilbereich n
INV,n:
Invertiert Teilbereich n
PAGE,n:
Wechselt Speicherbank des Bildschirms

n=0 für Bank 1 (16384 - 32767)
n=1 für Bank 3 (49152 - 65535)

Anweisungen zum Eintippen:

Listing 1 eintippen und abspeichern mit:
SAVE "PAINT"
Listing 2 eintippen, irgendwo mit "CREATOR" abspeichern, mit RUN starten und den erzeugten Code direkt hinter "PAINT" abspeichern lassen.
Listing 3 eintippen und hinter CODE mit:
SAVE "MASTER" abspeichern
Zum Einladen RUN "PAINT" eingeben. (Th. Fippl)

Listing 1

```
100 DEFINT a-z
                        [553]
110 SYMBOL AFTER 0                         [1116]
120 hm=UNT(HIMEM+1)                        [517]
130 MEMORY &94FF                           [782]
140 OPENOUT"dummy                          [1022]
150 MEMORY &3FFB                           [762]
160 CLOSEOUT                               [902]
170 POKE &9400,hm AND &FF                  [1306]
180 POKE &9401,PEEK(@hm+1)                 [1738]
190 FOR n=72 TO 75                         [1145]
200 KEY DEF n,1,n+168                      [1154]
210 NEXT                                   [350]
220 KEY DEF 76,1,224                       [1015]
230 KEY DEF 66,0,0,0                       [929]
240 KEY 139,CHR$(&FF)                      [1497]
250 SPEED KEY 12,1                         [1212]
260 LOAD"!code",&9500                      [663]
270 CALL &9514                             [682]
280 RUN"!master                            [487]
```

Listing 2

```
100 DEFINT a-z:m=&9500:MEMORY m-1          [1515]
110 FOR z=200 TO 272                       [1112]
120 s=0:FOR m=m TO m+7                     [1286]
130 READ b$:b=VAL("&"+b$)                  [1052]
140 POKE m,b:s=(s XOR b)+b                 [1182]
150 NEXT:READ sum                          [1332]
160 IF s-sum THEN PRINT"Error in"z:STOP    [1529]
170 NEXT:PRINT                             [811]
180 SAVE"code",b,&9500,584:END             [1587]
190 :                                      [174]
200 DATA eb,22,0b,97,c9,21,00,80,1386      [1431]
201 DATA 18,04,eb,11,00,80,01,fa,562       [1427]
202 DATA 00,ed,b0,c9,21,1d,95,01,1036      [1516]
203 DATA 21,95,c3,d1,bc,00,00,00,1156      [2336]
204 DATA 00,41,95,c3,6b,95,c3,7f,534       [1730]
205 DATA 95,c3,96,95,c3,90,95,c3,736       [1114]
206 DATA ae,95,c3,f9,95,c3,2b,96,1592      [2146]
207 DATA c3,42,96,c3,48,96,c3,d6,2146      [1717]
208 DATA 96,53,48,4f,d7,4b,45,45,920       [1740]
209 DATA d0,49,4e,d6,50,43,4c,d3,1194      [1503]
210 DATA 5a,4f,4f,cd,52,45,44,55,1050      [1765]
211 DATA 43,c5,53,57,41,d0,50,41,852       [968]
212 DATA 47,c5,46,49,4c,cc,47,50,1088      [991]
213 DATA 45,ce,00,cd,ed,96,eb,c5,1562      [1866]
214 DATA d5,06,00,ed,b0,e3,cd,26,1552      [1367]
215 DATA bc,eb,e1,c1,10,f1,c9,cd,1512      [1558]
216 DATA ed,96,c5,e5,06,00,ed,b0,496       [917]
217 DATA e1,cd,26,bc,c1,10,f3,c9,1566      [1677]
218 DATA dd,21,3f,97,18,04,dd,21,944       [1778]
219 DATA 42,97,cd,ed,96,c5,e5,cd,780       [1010]
220 DATA 46,97,23,0d,20,f9,e1,cd,772       [875]
221 DATA 26,bc,c1,10,f0,c9,cd,21,1548      [1348]
222 DATA 97,5f,3e,02,93,fe,01,17,1270      [1101]
223 DATA 5f,c5,48,05,cd,c4,95,c1,1288      [1499]
224 DATA 0d,20,f6,c9,c5,7e,cb,26,1444      [2143]
225 DATA a2,4f,cb,39,b1,10,fb,cd,1288      [1523]
226 DATA d8,95,c1,0d,20,ee,23,c9,978       [1941]
227 DATA e5,d5,57,0e,08,2a,02,94,906       [1427]
228 DATA e5,43,e5,72,23,10,fc,e1,1226      [2018]
229 DATA cd,26,bc,43,0d,20,f3,e1,1186      [1347]
230 DATA 23,10,fd,22,02,94,d1,e1,1068      [1934]
231 DATA c9,cd,21,97,57,3e,02,92,926       [2120]
232 DATA fe,01,17,e5,c5,f5,cd,12,1484      [1048]
233 DATA 96,f1,c1,e1,72,23,0d,20,1062      [1159]
234 DATA f2,c9,48,47,16,00,2a,02,716       [1710]
235 DATA 94,c5,7e,a3,cb,22,b2,57,1820      [767]
236 DATA 23,10,fd,c1,0d,20,f2,22,1036      [2093]
237 DATA 02,94,c9,cd,ed,96,c5,e5,1702      [2742]
238 DATA 1a,46,77,78,12,23,13,0d,448       [1766]
239 DATA 20,f6,e1,cd,26,bc,c1,10,1086      [1409]
240 DATA ed,c9,7b,2f,0f,c3,08,bc,1188      [2079]
241 DATA cd,c6,bb,cd,11,bc,28,06,1292      [1606]
242 DATA 30,08,cb,3a,cb,1b,cb,3a,964       [1797]
243 DATA cb,1b,cb,3c,cb,1d,d5,7d,1382      [1479]
244 DATA dd,6f,dd,67,cd,c8,96,30,1626      [898]
245 DATA 06,cd,7a,96,13,18,f5,d1,1484      [1099]
246 DATA 1b,cd,c8,96,d0,cd,7a,96,758       [1600]
247 DATA 18,f6,e5,2a,04,94,e5,cd,1242      [981]
248 DATA 29,bc,dd,2c,cd,9e,96,38,1430      [1914]
249 DATA f6,dd,7c,dd,6f,e1,cd,26,1106      [1842]
250 DATA bc,dd,2d,cd,9e,96,38,f6,1246      [1905]
251 DATA e1,dd,7c,dd,6f,c9,cd,aa,1328      [2073]
252 DATA 96,d0,7e,a1,37,cc,e2,96,1900      [1037]
253 DATA 3f,c9,dd,7d,fe,c8,d0,d5,1730      [2193]
254 DATA e5,eb,cd,11,bc,11,a0,00,1002      [2239]
255 DATA 38,08,11,40,01,28,03,11,316       [971]
256 DATA 80,02,a7,ed,52,e1,d1,c9,1502      [1883]
257 DATA e5,d5,cd,1d,bc,d1,22,04,1130      [1126]
258 DATA 94,cd,9e,96,e1,c9,7b,cd,1026      [1148]
259 DATA de,bb,7b,cd,2c,bc,32,06,1414      [925]
260 DATA 94,c9,3a,06,94,47,7e,a8,1112      [1518]
261 DATA b1,a9,a8,77,c9,7b,e6,07,1636      [610]
262 DATA 87,4f,87,87,81,4f,cb,3b,1076      [2049]
263 DATA cb,3b,7b,e6,fe,21,11,97,1340      [1369]
264 DATA 85,6f,7e,23,66,81,6f,8c,1102      [1206]
265 DATA 95,67,11,00,80,01,0a,19,446       [603]
266 DATA c9,00,c0,f0,c8,e0,d1,d0,1444      [846]
267 DATA da,c0,e3,b0,ec,a0,f5,90,1932      [1502]
268 DATA fe,21,00,40,22,02,94,26,510       [1688]
269 DATA 80,cd,11,bc,11,55,aa,01,758       [1424]
270 DATA fa,02,d8,11,11,88,06,04,520       [1190]
271 DATA c8,11,01,80,06,08,c9,36,610       [1520]
272 DATA 00,c9,7e,2f,77,c9,dd,e9,1408      [1098]
```

Listing 3

```
100 '***************                       [528]
110 '*             *                       [175]
120 '* PAINT-MASTER *                      [343]
130 '*   (c) 1985  *                       [542]
140 '* by Th. Fippl *                      [723]
150 '*  6336 Solms *                       [754]
160 '*             *                       [175]
170 '***************                       [528]
180 '                                      [117]
190 '                                      [117]
1000 CLEAR:DEG                             [381]
1010 DEFSTR a-k:DEFINT m-z                 [955]
1020 FOR n=0 TO 2:x=14-2*n                 [1254]
1030 FOR m=1 TO x:READ y                   [882]
1040 cmd(n)=cmd(n)+CHR$(y)                 [886]
1050 NEXT m,n:DIM vi(15)                   [1257]
1060 FOR n=0 TO 15:READ m                  [1174]
1070 INK n,m:vi(n)=m:NEXT                  [1664]
1080 BORDER 2:ZONE 42                      [1436]
1090 xa=320:ya=200:xb=xa:yb=ya             [2874]
1100 md=1:sz=2:pz=3:stx=4:sty=4:pn=1:vb=2  [3120]
1110 mtx=UNT(PEEK(&9400)+256*PEEK(&9401))  [2002]
1120 :                                     [174]
1130 DATA &47,&44,&52,&59,&41,&43,&45      [1226]
1140 DATA &51,&46,&4a,&10,&4f,&58,&48      [1245]
1150 DATA &20,&50,&49,&42,&54,&56          [1202]
1160 DATA &4b,&57,&4d,&53,&4c,&55          [932]
1170 DATA &f0,&f1,&f2,&f3,&e0              [1336]
1180 DATA &7f,&0d,&4e,&09,&5a              [1514]
1190 DATA 1,24,15,6,26,0,2,11,8,10,12,14,1 [2406]
6,18,20,22
1200 :                                     [174]
1210 ON ERROR GOTO 5570                    [1344]
1220 MODE md:|PAGE,0                       [1031]
1230 GOSUB 1390                            [933]
1240 i=UPPER$(INKEY$)                      [1461]
1250 IF i=""GOTO 1240                      [485]
1260 GOSUB 1390                            [933]
1270 IF i=CHR$(&FF)GOTO 5470               [1333]
1280 :                                     [174]
1290 m=INSTR(cmd(0),i)                     [515]
1300 IF m THEN z=m:GOTO 1230               [2745]
1310 m=INSTR(cmd(2),i)                     [355]
1320 ON m GOSUB 2000,2040,2080,2120,2160,3 [3857]
860,3720,3650,3760,3320
1330 m=INSTR(cmd(1),i)                     [243]
1340 IF m=0 GOTO 1230                      [1011]
1350 |PAGE,1:MODE 1:LOCATE 2,2             [1948]
1360 ON m GOSUB 3930,4170,4240,4360,4430,4 [3623]
```

```
510,4680,4780,4900,4990,5270,5370
1370 GOTO 1220                              [359]
1380 :                                      [174]
1390 PRINT CHR$(23)CHR$(1);                 [1919]
1400 IF z=2 THEN MOVE xb,yb:DRAW xa,ya,pn   [3047]
1410 MOVE 0,ya:DRAWR 640,0,1                [2137]
1420 MOVE xa,0:DRAWR 0,400                  [1984]
1430 MOVE xb-8,yb-8:DRAWR 16,16             [517]
1440 MOVE xb+8,yb-8:DRAWR-16,16             [1305]
1450 PRINT CHR$(23)CHR$(wr);                [2028]
1460 RETURN                                 [555]
1470 :                                      [174]
1480 m=INT(xa/80)                           [649]
1490 n=INT((399-ya)/50)                     [1339]
1500 num=n*8+m:RETURN                       [1172]
1510 :                                      [174]
1520 RESTORE 1620:LOCATE 18,3               [1609]
1530 PRINT CHR$(24)" Farbpalette "CHR$(24)  [4105]
1540 FOR n=0 TO 12                          [564]
1550 READ a,b:LOCATE 12,6+n                 [1428]
1560 PRINT USING"## "+a;n;                  [898]
1570 PRINT TAB(26)USING"## "+b;n+13         [1178]
1580 NEXT:LOCATE 26,19                      [1053]
1590 PRINT"26 h. Weiss                      [1879]
1600 RETURN                                 [555]
1610 :                                      [174]
1620 DATA Schwarz,Weiss,Blau,Pastellblau    [1192]
1630 DATA Hellblau,Orange,Rot,Rosa          [1123]
1640 DATA Magenta,Pastellmag.,h. Violett    [2309]
1650 DATA Hellgruen,Hellrot,Seegruen        [1931]
1660 DATA Purpur,h. Blaugr.,h. Magenta      [1723]
1670 DATA Limonengr.,Gruen,Pastellgr.       [2531]
1680 DATA Blaugruen,Pa. blaugr.,Himmelblau  [3042]
1690 DATA Hellgelb,Gelb,Pastellgelb         [1265]
1700 :                                      [174]
1710 x=60:y=340:p=0:TAG:|GPEN,1             [2556]
1720 FOR n=0 TO 15:FOR m=0 TO 15            [2101]
1730 MOVE 66+m*32,336-n*20                  [1782]
1740 PRINT CHR$(n*16+m);                    [1484]
1750 NEXT m,n:TAGOFF:|GPEN,2                [1304]
1760 PRINT CHR$(23)CHR$(1);                 [1919]
1770 RETURN                                 [555]
1780 :                                      [174]
1790 GOSUB 1960                             [851]
1800 i=INKEY$:IF i=""GOTO 1800              [1092]
1810 m=INSTR(LEFT$(cmd(2),7),i)             [1508]
1820 GOSUB 1960                             [851]
1830 IF m>4 THEN RETURN                     [1157]
1840 ON m GOSUB 1870,1890,1910,1930         [2562]
1850 GOTO 1790                              [341]
1860 :                                      [174]
1870 y=y+20:IF y>340 THEN y=40:q=240 ELSE   [2883]
q=-16
1880 p=p+q:RETURN                           [884]
1890 y=y-20:IF y<40 THEN y=340:q=-240 ELSE  [3569]
 q=16
1900 p=p+q:RETURN                           [884]
1910 x=x-32:IF x<60 THEN x=540:q=15 ELSE q  [2345]
=-1
1920 p=p+q:RETURN                           [884]
1930 x=x+32:IF x>540 THEN x=60:q=-15 ELSE   [4077]
q=1
1940 p=p+q:RETURN                           [884]
1950 :                                      [174]
1960 MOVE x,y:DRAWR 26,0                    [920]
1970 DRAWR 0,-22:DRAWR-26,0                 [1427]
1980 DRAWR 0,22:RETURN                      [1586]
1990 '                                      [117]
2000 '*** CURSOR UP ***                     [847]
2010 '                                      [117]
2020 ya=MIN(ya+2*sty,399):RETURN            [2251]
2030 '                                      [117]
2040 '*** CURSOR DOWN ***                   [1410]
2050 '                                      [117]
2060 ya=MAX(ya-2*sty,0):RETURN              [3110]
2070 '                                      [117]
2080 '*** CURSOR LEFT ***                   [1168]
2090 '                                      [117]
2100 xa=MAX(xa-sz*stx,0):RETURN             [3043]
2110 '                                      [117]
2120 '*** CURSOR RIGHT ***                  [1062]
2130 '                                      [117]
2140 xa=MIN(xa+sz*stx,639):RETURN           [3437]
2150 '                                      [117]
2160 '*** SETUP ***                         [649]
2170 '                                      [117]
2180 |GPEN,pn                               [854]
2190 ON z GOTO 2220,2260,2310,2370,2430,26  [3907]
10,2690,2770,2940,2980,3050,3120,3160,3200
```

```
,3270
2200 RETURN                                    [555]
2210 '                                         [117]
2220 '*** PLOT ***                             [786]
2230 '                                         [117]
2240 PLOT xa,ya:RETURN                         [1618]
2250 '                                         [117]
2260 '*** DRAW ***                             [1284]
2270 '                                         [117]
2280 MOVE xb,yb:DRAW xa,ya                     [1764]
2290 xb=xa:yb=ya:RETURN                        [1376]
2300 '                                         [117]
2310 '*** REC ***                              [974]
2320 '                                         [117]
2330 MOVE xb,yb:DRAW xb,ya                     [1723]
2340 DRAW xa,ya:DRAW xa,yb                     [1387]
2350 DRAW xb,yb:RETURN                         [653]
2360 '                                         [117]
2370 '*** BOX ***                              [759]
2380 '                                         [117]
2390 FOR y=yb TO ya STEP SGN(ya-yb)*2          [1372]
2400 MOVE xb,y:DRAW xa,y:NEXT                  [1305]
2410 RETURN                                    [555]
2420 '                                         [117]
2430 '*** CUBE ***                             [836]
2440 '                                         [117]
2450 rflag=rflag XOR 1                         [1513]
2460 ON rflag GOTO 2590                        [751]
2470 x=xc-xb:y=yc-yb                           [1238]
2480 IF x<0 THEN x1=MIN(xa,xb):x2=MAX(xa,x     [3493]
b)ELSE x1=MAX(xa,xb):x2=MIN(xa,xb)
2490 IF y<0 THEN y1=MAX(ya,yb):y2=MIN(ya,y     [4338]
b)ELSE y1=MIN(ya,yb):y2=MAX(ya,yb)
2500 GOSUB 2330                                [857]
2510 MOVE x1,y1                                [432]
2520 DRAWR x,y                                 [197]
2530 DRAWR 0,y2-YPOS+y                         [738]
2540 DRAWR-x,-y                                [780]
2550 DRAWR x,y                                 [197]
2560 DRAWR x2-XPOS+x,0                         [536]
2570 DRAWR-x,-y                                [780]
2580 RETURN                                    [555]
2590 xc=xa:yc=ya:RETURN                        [2108]
2600 '                                         [117]
2610 '*** CIRCLE ***                           [908]
2620 '                                         [117]
2630 rx=ABS(xa-xb):ry=ABS(ya-yb)               [909]
2640 FOR l=0 TO 366 STEP 360/SQR(rx*ry+1)      [1333]
2650 x=xb+rx*SIN(l):y=yb+ry*COS(l)             [1861]
2660 IF l>0 THEN DRAW x,y ELSE PLOT x,y        [3170]
2670 NEXT:RETURN                               [940]
2680 '                                         [117]
2690 '*** SPOT ***                             [640]
2700 '                                         [117]
2710 rx=ABS(xa-xb):ry=ABS(ya-yb)               [909]
2720 FOR l=0 TO 180 STEP 90/(ry+1)             [1429]
2730 x=rx*SIN(l):y=yb+ry*COS(l)                [4372]
2740 MOVE xb-x,y:DRAWR 2*x,0                   [1094]
2750 NEXT:RETURN                               [940]
2760 '                                         [117]
2770 '*** LARGE ***                            [1121]
2780 '                                         [117]
2790 x=(xb-xa)\8:y=(ya-yb)\8                   [1071]
2800 FOR n=1 TO LEN(k)                         [604]
2810 l=mtx+ASC(MID$(k,n))*8                    [1314]
2820 FOR ma=0 TO 7                             [808]
2830 b=BIN$(PEEK(l+ma),8)                      [2203]
2840 FOR mb=1 TO 8                             [846]
2850 IF MID$(b,mb,1)="1"THEN GOSUB 2880        [2481]
2860 NEXT mb,ma,n:RETURN                       [1494]
2870 :                                         [174]
2880 xc=xa+x*((n-1)*8+mb-1)                    [2350]
2890 yc=ya-y*ma                                [859]
2900 FOR yd=yc TO yc-y STEP 2*SGN(-y)          [2393]
2910 MOVE xc,yd:DRAWR x,0                      [1039]
2920 NEXT:RETURN                               [940]
2930 '                                         [117]
2940 '*** FILL ***                             [1254]
2950 '                                         [117]
2960 MOVE xa,ya:|FILL:RETURN                   [2346]
2970 '                                         [117]
2980 '*** SPRAY ***                            [517]
2990 '                                         [117]
3000 x=xa-xb:y=ya-yb                           [1654]
3010 FOR n=0 TO ABS(x*y)/250/sz                [1685]
3020 PLOT xb+x*RND,yb+y*RND                    [2169]
3030 NEXT:RETURN                               [940]
3040 '                                         [117]
3050 '*** CLG ***                              [769]
3060 '                                         [117]
3070 x1=MIN(xa,xb):x2=MAX(xa,xb)               [2600]
3080 y1=MAX(ya,yb):y2=MIN(ya,yb)               [2827]
3090 ORIGIN 0,0,x1,x2,y1,y2:CLG                [1094]
3100 ORIGIN 0,0,0,639,399,0:RETURN             [1422]
3110 '                                         [117]
3120 '*** PCLS ***                             [698]
3130 '                                         [117]
3140 GOSUB 1480:|PCLS,num:RETURN               [1281]
3150 '                                         [117]
3160 '*** INVERT ***                           [786]
3170 '                                         [117]
3180 GOSUB 1480:|INV,num:RETURN                [1277]
3190 '                                         [117]
3200 '*** COPY ***                             [1157]
3210 '                                         [117]
3220 x=xa:y=ya:xa=xb:ya=yb                     [1053]
3230 GOSUB 1480:|KEEP,num                      [1312]
3240 xa=x:ya=y:GOSUB 1480                      [2158]
3250 |SHOW,num:RETURN                          [1786]
3260 '                                         [117]
3270 '*** PRINT ***                            [1131]
3280 '                                         [117]
3290 TAG:MOVE xa,ya:PRINT k;                   [1603]
3300 TAGOFF:RETURN                             [1224]
3310 '                                         [117]
3320 '*** ZOOM ***                             [1301]
3330 '                                         [117]
3340 GOSUB 1480:|KEEP,num                      [1312]
3350 x=INT((xa-m*80)/sz)+1                     [1657]
3360 y=INT((399-ya-n*50)/2)+1                  [1594]
3370 |PAGE,1:MODE md:|ZOOM                     [2111]
3380 PEN pn:PRINT CHR$(7);                     [1020]
3390 EVERY 15 GOSUB 3520                       [1359]
3400 i=INKEY$:IF i=""GOTO 3400                 [1220]
3410 m=REMAIN(0):ON vflag GOSUB 3520           [1266]
3420 IF i=CHR$(13)GOTO 3470                    [1083]
3430 m=INSTR(LEFT$(cmd(2),6),i)                [1651]
3440 ON m GOSUB 3550,3560,3570,3580,3600,3     [2538]
620
3450 GOSUB 3520:GOTO 3390                      [1077]
3460 :                                         [174]
3470 PRINT CHR$(7);                            [1175]
3480 i=INKEY$:IF i=""GOTO 3480                 [1141]
3490 IF i<>CHR$(13)GOTO 3390                   [1409]
3500 |REDUCE:|PAGE,0:|SHOW,num                 [1619]
3510 PEN 1:RETURN                              [1269]
3520 LOCATE x,y:CALL &BB8A                     [898]
3530 vflag=vflag XOR 1:RETURN                  [599]
3540 :                                         [174]
3550 y=MAX(y-1,1):RETURN                       [1681]
3560 y=MIN(y+1,25):RETURN                      [1853]
3570 x=MAX(x-1,1):RETURN                       [2922]
3580 xwin=-19*(md=0)-39*(md=1)-79*(md=2)       [2886]
3590 x=MIN(x,xwin)+1:RETURN                    [2164]
3600 LOCATE x,y:PRINT CHR$(143);               [1087]
3610 RETURN                                    [555]
3620 LOCATE x,y:PRINT CHR$(32);                [1156]
3630 RETURN                                    [555]
3640 '                                         [117]
3650 '*** SWAP ***                             [755]
3660 '                                         [117]
3670 GOSUB 1480:|KEEP,num:nm=num               [2313]
3680 x=xa:y=ya:xa=xb:ya=yb:xb=x:yb=y           [1215]
3690 GOSUB 1480:|SWAP,num:|SHOW,nm             [1890]
3700 RETURN                                    [555]
3710 '                                         [117]
3720 '*** MOVE ***                             [1219]
3730 '                                         [117]
3740 xb=xa:yb=ya:RETURN                        [1376]
3750 '                                         [117]
3760 '*** FIND ***                             [1095]
3770 '                                         [117]
3780 GOSUB 1480                                [819]
3790 x=m*80:y=399-n*50                         [1025]
3800 PRINT CHR$(23)CHR$(1);                    [1919]
3810 GOSUB 3820:CALL &BB06                     [1754]
3820 MOVE x,y:DRAWR 0,-49                      [944]
3830 DRAWR 79,0:DRAWR 0,49                     [1021]
3840 DRAWR-79,0:RETURN                         [759]
3850 '                                         [117]
3860 '*** ERASE ***                            [767]
3870 '                                         [117]
3880 PRINT CHR$(23)CHR$(0);                    [1923]
3890 POKE &96A5,&C4:|GPEN,0                    [2072]
3900 GOSUB 2190:|GPEN,pn                       [1503]
3910 POKE &96A5,&CC:RETURN                     [1393]
3920 '                                         [117]
```

```
3930 '*** HELP ***                          [692]
3940 '                                      [117]
3950 RESTORE 4050                           [924]
3960 FOR y=2 TO 22 STEP 2                   [1274]
3970 FOR x=2 TO 28 STEP 13                  [956]
3980 READ a,b:LOCATE x,y                    [1587]
3990 PRINT CHR$(24)RIGHT$(".."+a,3)CHR$(24  [3018]
)" - "b;
4000 NEXT x,y:LOCATE 7,24                   [850]
4010 WHILE INKEY$<>"":WEND                  [1786]
4020 PRINT"Press any key to continue        [2050]
4030 CALL &BB06:RETURN                      [1826]
4040 :                                      [174]
4050 DATA G,PLOT,Z,ZOOM,COP,SETUP           [1486]
4060 DATA D,DRAW,CLR,CLG,ENT,MOVE           [1180]
4070 DATA R,REC,O,PCLS,TAB,FIND             [1601]
4080 DATA Y,BOX,X,INVERT,W,WRITE            [1702]
4090 DATA A,CUBE,N,SWAP,DEL,ERASE           [2238]
4100 DATA C,CIRCLE,H,COPY,M,MODE            [1535]
4110 DATA E,SPOT,P,PEN,ent,EXIT             [1767]
4120 DATA Q,LARGE,I,INK,S,SAVE              [2543]
4130 DATA F,FILL,B,BORDER,L,LOAD            [1027]
4140 DATA J,SPRAY,V,SYMBOL,U,MERGE          [1622]
4150 DATA T,TEXT,K,STEP,SPC,HELP            [1139]
4160 '                                      [117]
4170 '*** PEN ***                           [616]
4180 '                                      [117]
4190 PRINT"Pen:"pn                          [879]
4200 LOCATE 2,4:INPUT"Pen";pn               [2223]
4210 IF pn<0 OR pn>pz GOTO 4200             [1220]
4220 |GPEN,pn:RETURN                        [1184]
4230 '                                      [117]
4240 '*** INK ***                           [321]
4250 '                                      [117]
4260 PRINT"Ink Table                        [864]
4270 FOR p=0 TO pz:LOCATE 3,4+p             [1511]
4280 IF pn=p THEN PRINT CHR$(24);:xflag=1   [2211]
4290 PRINT USING"## -"+STR$(vi(p));p        [1473]
4300 IF xflag THEN PRINT CHR$(24):xflag=0   [2329]
4310 NEXT:GOSUB 1520                        [974]
4320 LOCATE 2,22:INPUT"Ink [Nr,f] ";n,m     [2686]
4330 IF n<0 OR m<0 OR n>pz OR m>26 GOTO 43  [2540]
20
4340 vi(n)=m:INK n,m:RETURN                 [2501]
4350 '                                      [117]
4360 '*** BORDER ***                        [974]
4370 '                                      [117]
4380 PRINT"Border:"vb:GOSUB 1520            [2835]
4390 LOCATE 2,4:INPUT"Border";vb            [2705]
4400 IF vb<0 OR vb>26 GOTO 4390             [2186]
4410 BORDER vb:RETURN                       [1189]
4420 '                                      [117]
4430 '*** TEXT ***                          [610]
4440 '                                      [117]
4450 PRINT"Text: ";:GOSUB 1710:k=""         [2509]
4460 GOSUB 1790:IF m=7 THEN z=15:RETURN     [2162]
4470 IF m=5 THEN k=k+CHR$(p):PRINT CHR$(1)  [3836]
CHR$(p);:GOTO 4460
4480 m=LEN(k)-1:IF m>=0 THEN k=LEFT$(k,m):  [3438]
PRINT CHR$(8)" "CHR$(8);
4490 GOTO 4460                              [317]
4500 '                                      [117]
4510 '*** SYMBOL ***                        [563]
4520 '                                      [117]
4530 LOCATE 14,3                            [674]
4540 PRINT CHR$(24)" Zeichensatz "CHR$(24)  [2950]
4550 GOSUB 1710                             [869]
4560 GOSUB 1790:IF m>5 GOTO 4560            [2163]
4570 :                                      [174]
4580 MODE 1:LOCATE 2,2                      [1337]
4590 PRINT CHR$(1)CHR$(p)"  Code:"p         [2620]
4600 WINDOW#1,2,9,4,12                      [1257]
4610 PEN#1,1:PAPER#1,2:CLS#1                [916]
4620 LINE INPUT#1,a                         [1453]
4630 a=LEFT$(a+STRING$(64,"0"),64)          [2695]
4640 l=mtx+p*8:FOR n=0 TO 7                 [833]
4650 POKE l+n,VAL("&x"+MID$(a,n*8+1,8))     [1431]
4660 NEXT:RETURN                            [940]
4670 '                                      [117]
4680 '*** STEP ***                          [914]
4690 '                                      [117]
4700 PRINT"X-Step:"stx                      [970]
4710 PRINT" Y-Step:"sty                     [1290]
4720 LOCATE 2,5:INPUT"X-Step ";stx          [2556]
4730 IF stx<1 OR stx>80 GOTO 4720           [883]
4740 LOCATE 2,7:INPUT"Y-Step ";sty          [1176]
4750 IF sty<1 OR sty>80 GOTO 4740           [2293]
4760 RETURN                                 [555]
4770 '                                      [117]
4780 '*** WRITE ***                         [947]
4790 '                                      [117]
4800 RESTORE 4880                           [759]
4810 FOR n=0 TO 3:READ a                    [916]
4820 LOCATE 2,2+n*2                         [870]
4830 IF n=wr THEN PRINT CHR$(24)n"- "a" "C  [3377]
HR$(24)ELSE PRINT n"- "a
4840 NEXT                                   [350]
4850 LOCATE 2,12:INPUT wr                   [2265]
4860 IF wr<0 OR wr>3 GOTO 4850              [1981]
4870 RETURN                                 [555]
4880 DATA NORMAL,XOR,AND,OR Mode            [1201]
4890 '                                      [117]
4900 '*** MODE ***                          [1442]
4910 '                                      [117]
4920 PRINT"Mode:"md                         [1820]
4930 LOCATE 2,4:INPUT"Mode ";md             [3223]
4940 IF md<0 OR md>2 GOTO 4930              [379]
4950 xa=320:ya=200:xb=xa:yb=ya              [2874]
4960 sz=2^(2-md):pz=2^sz-1                  [1587]
4970 |PAGE,0:RETURN                         [1332]
4980 '                                      [117]
4990 '*** SAVE ***                          [588]
5000 '                                      [117]
5010 INPUT"Name des Bildes ";b              [1951]
5020 IF LEN(b)<1 OR LEN(b)>8 GOTO 5010      [2110]
5030 PRINT,CHR$(24);1;CHR$(24);             [1664]
5040 PRINT" Gesamtbild"                     [1658]
5050 PRINT,CHR$(24);2;CHR$(24);             [2243]
5060 PRINT" Teilbild":PRINT                 [995]
5070 i=INKEY$:IF i<"1"OR i>"2"GOTO 5070     [1918]
5080 MODE md:|PAGE,0                        [1031]
5090 ON VAL(i)GOSUB 5110,5150               [1746]
5100 |PAGE,1:RETURN                         [1458]
5110 POKE 65518,md:POKE 65519,vb            [1307]
5120 FOR n=0 TO 15                          [577]
5130 POKE 65520+n,vi(n):NEXT                [1130]
5140 SAVE"!"+b+".gsb",b,49152,16384:RETURN  [2130]
5150 GOSUB 1480:s=n:r=m:p=16384             [2009]
5160 xa=xb:ya=yb:GOSUB 1480                 [1561]
5170 nv=MIN(n,s):nb=MAX(n,s)                [1902]
5180 mv=MIN(m,r):mb=MAX(m,r)                [453]
5190 FOR y=nv TO nb                         [1044]
5200 FOR x=mv TO mb                         [1198]
5210 |KEEP,y*8+x:CALL &9505,p               [1964]
5220 p=p+250:NEXT x,y                       [1875]
5230 POKE 16380,nv:POKE 16381,nb            [1610]
5240 POKE 16382,mv:POKE 16383,mb            [1573]
5250 SAVE"!"+b+".tlb",b,16380,p-16380:RETU  [3282]
RN
5260 '                                      [117]
5270 '*** LOAD ***                          [1291]
5280 '                                      [117]
5290 INPUT"Name des Gesamtbildes ";b        [1890]
5300 IF LEN(b)<1 OR LEN(b)>8 GOTO 5290      [2118]
5310 LOAD b+".gsb",&C000                    [1482]
5320 md=PEEK(65518):sz=2^(2-md):pz=2^sz-1   [2569]
5330 vb=PEEK(65519):BORDER vb               [2378]
5340 FOR n=0 TO 15:vi(n)=PEEK(65520+n)      [2285]
5350 INK n,vi(n):NEXT:RETURN                [3266]
5360 '                                      [117]
5370 '*** MERGE ***                         [1249]
5380 '                                      [117]
5390 INPUT"Name des Teilbildes ";b          [2264]
5400 IF LEN(b)<1 OR LEN(b)>8 GOTO 5390      [2298]
5410 |PAGE,0:p=16384:LOAD"!"+b+".tlb",1638  [2388]
0
5420 FOR y=PEEK(16380)TO PEEK(16381)        [2060]
5430 FOR x=PEEK(16382)TO PEEK(16383)        [2323]
5440 CALL &950A,p:|SHOW,y*8+x               [1425]
5450 p=p+250:NEXT x,y:|PAGE,1:RETURN        [2924]
5460 '                                      [117]
5470 '*** EXIT ***                          [566]
5480 '                                      [117]
5490 |PAGE,1:MODE 1:LOCATE 2,2              [1948]
5500 PRINT"Beenden ? ";:CALL &BB8A          [2229]
5510 i=INKEY$:IF i=""GOTO 5510              [1159]
5520 CALL &BB8D:LOCATE 2,4                  [1301]
5530 IF i<CHR$(&FF)GOTO 1220                [2140]
5540 PRINT"Continue: 'GOTO 1200'            [2470]
5550 PRINT:END                              [738]
5560 '                                      [117]
5570 '*** ON ERROR ***                      [707]
5580 '                                      [117]
5590 |PAGE,1:MODE 1:LOCATE 2,2              [1948]
5600 PRINT"Error"ERR"in Line"ERL            [2980]
5610 LOCATE 2,4:GOSUB 4010                  [2262]
5620 GOTO 1220                              [359]
```

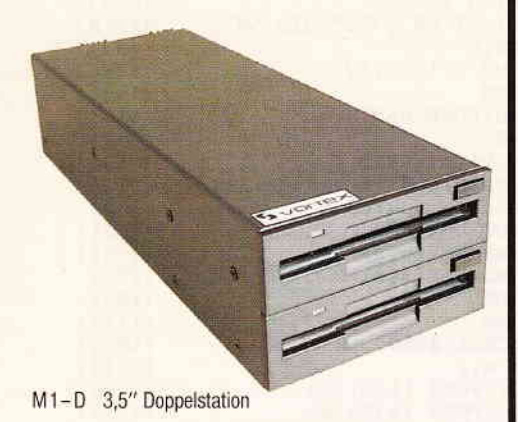
für CPC 464 - 664 - 6128:

DIE VORTEX

Sie haben noch keine Disk Station? Dann steigen Sie doch gleich richtig ein!

Die Kraftwerke: vortex Floppy Disk Station F1 und M1

708 KiloByte oder 1.4 MegaByte auf Standard 5,25" oder 3,5" Disketten – damit haben Ihre Programme genügend „Auslauf".



M1-D 3,5" Doppelstation

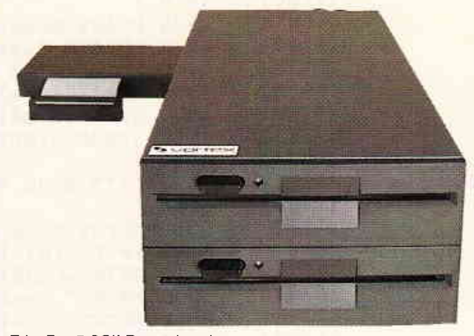
Fantastische Leistungsdaten:

5,25" oder 3,5" Qualitätslaufwerke mit 2 Schreib-/Leseköpfen und insgesamt 160 Spuren.
Die Geräte F1-S (5,25", 708 KB) und M1-S (3,5", 708 KB) können einfachst durch Einbau eines weiteren Laufwerks zu den Geräten F1-D (5,25", 1.4 MB) und M1-D (3,5", 1.4 MB) erweitert werden.
Pro Diskette können bis zu 128 Dateien angelegt werden.
Es kann zusätzlich ein original Schneider 3" Laufwerk angeschlossen werden. Damit können Sie nach Belieben zwischen 3" und 5,25"/3,5" kopieren.

Es ist auch ein „gemischtes Doppel" erhältlich: vortex FM1: ein 5,25" Laufwerk und ein 3,5" Laufwerk:



FM1 5,25"/3,5" Doppelstation



F1-D 5,25" Doppelstation

Zwei professionelle Betriebssysteme:

- CP/M 2.2 und CP/M plus (nur CPC 6128)
- Unter BASIC: VDOS 2.1 – das vortex Disk Operating System. Es ist das Beste ...

- Relative Dateiverwaltung, es können bis zu 16 Dateien gleichzeitig bearbeitet werden.
- Z80 Maschinensprache Monitor.
- Disketteneditor.
- Automatische Formaterkennung (System/Data Format) bei angeschlossenem 3" Laufwerk.
- Disketten formatieren unter BASIC.



Floppy Disk Controller

Intelligente Floppycontroller Elektronik

- Background ROM Nummer frei einstellbar. Damit wird sichergestellt, daß sich zwei Peripheriegeräte nicht „stören".

- Ausführliches deutsches Handbuch.

Und damit Sie nicht länger überlegen müssen:

vortex Floppy F1-S bzw. M1-S	DM **998,-***
vortex Floppy F1-D bzw. M1-D	DM **1498,-***

Alle Geräte sind sofort anschlußfertig.

*unverbindliche Preisempfehlung

Der Tophit für jeden CPC (464 + DDI-1, 664, 6128):

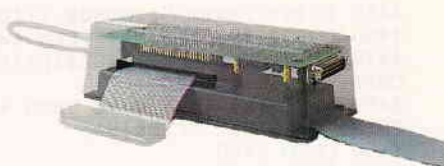
1 Megabyte = 758 DM

Damit präsentieren wir eine professionelle Systemlösung für alle die, die schon ein 3" Laufwerk haben, aber den Standard brauchen (F1-X = 5,25" bzw. M1-X = 3,5").
Unsere Formel lautet:
180 KB (3") + 708 KB (X-System) = 888 KB.



Das 3,5" X-Laufwerk

Das X(RS)-Modul erweitert den 3" Controller um die Leistungsdaten unseres VDOS 2.0 Betriebssystems.



Trickfoto: X-Modul (RS-Ausführung) und 3" Schneider Controller

- Das X-Laufwerk ist kein Zweitlaufwerk, sondern Ihr Systemlaufwerk.
- Die Befehle "ICPM,1" und "ICPM,2" ermöglichen es erstmals, CP/M von beiden Drives gleichberechtigt zu starten.
- Ihr 3" Laufwerk und das X-Drive unterscheiden sich zwar wesentlich in Datendichte und Datenformat, das Kopieren geschieht jedoch so reibungslos und glatt, als gäbe es diesen Unterschied nicht.
- Sie wählen per Kommando zwischen "AMSDOS" und "VDOS 2.0" als Betriebssystem – Sie verfügen über beides – nach Ihrer Wahl.
- Als herausragende Leistung bietet Ihnen VDOS 2.0:
  - 128 Directory Einträge
  - direkter relativer Dateizugriff unter

Wenn Sie keinen Händler in der Nähe haben, rufen Sie doch einfach an!

Das Software-Experiment

Folge 2: Das Spiel des Lebens

Leistungsfähige, grafisch orientierte Computersysteme gewinnen in der Industrie, Wissenschaft und Kunst zunehmend an Bedeutung. Die Entwicklung und Darstellung hochkomplexer Strukturen ist in vielen Fällen überhaupt erst durch den Einsatz bildverarbeitender Systeme möglich geworden. Und erstaunlich genug – oft liefern selbst rein zweckorientierte Forschungsprojekte ein Bildmaterial, das man eher einem genialen Künstler zutrauen möchte. Die in der letzten Folge behandelten "Chaos-Grafiken" stellen sicherlich eines der erstaunlichsten Beispiele in dieser Hinsicht dar.

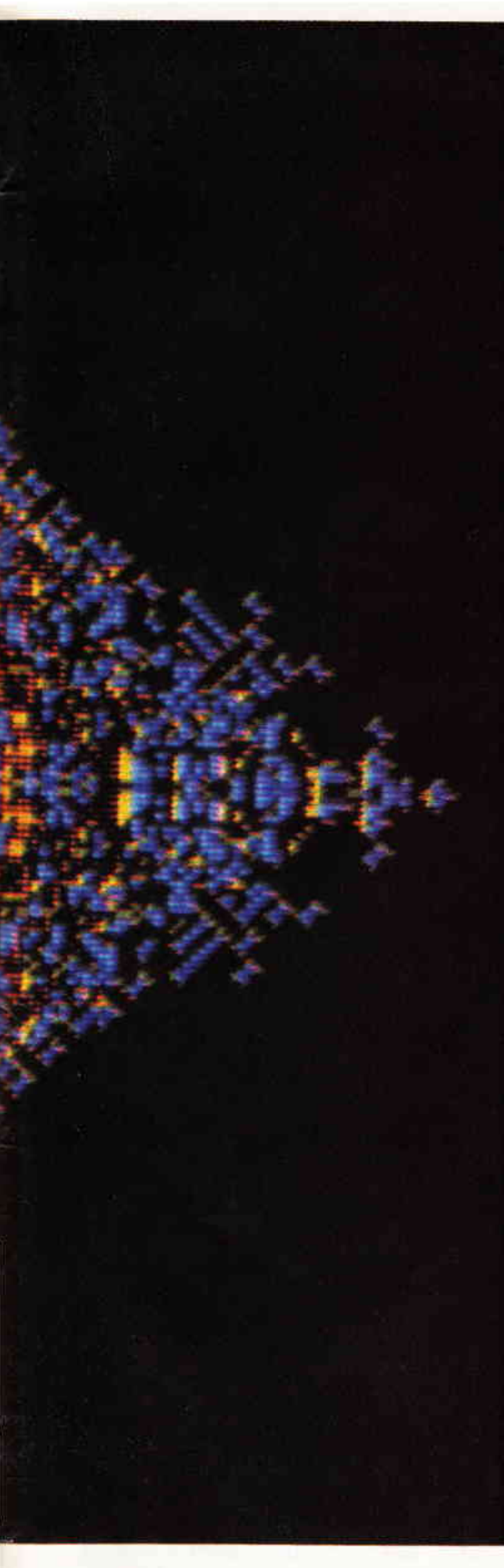
Doch bei aller Begeisterung für die Möglichkeiten der neuen Technologie sollte man nicht vergessen, daß sich in der Natur zahlreiche "Kreationen" finden lassen, neben denen die meisten Computergrafiken noch recht blaß aussehen. Haben Sie schon einmal Schneekristalle durch eine Lupe oder ein Mikroskop betrachtet? Falls nicht, so sollten Sie es bei nächster Gelegenheit nachholen, oder zumindest lesen, was der Schriftsteller Thomas Mann über dieses Erlebnis berichtet:

"Kleinodien, Ordenssterne, Brillantgaraffen, wie der getreueste Juwelier sie nicht reicher und minuziöser hätte herstellen können..., und unter den Myriaden von Zaubersternchen in ihrer untersichtigen, dem Menschenauge nicht zugedachten, heimlichen Kleinpracht war nicht eines dem anderen gleich."

Der unerschöpfliche Erfindungsgeist, mit dem die Natur milliardenfache Variationen des Themas "sechsstrahlige Symmetrie" erzeugt, erscheint nach wie vor als ein unbegreifliches Wunder – und doch ist es inzwischen mit Hilfe intelligent ausgedachter Computersimulationen gelungen, zumindest einige der Grundprinzipien dieser Naturvorgänge zu erforschen und zu verstehen. So konnte unter anderem gezeigt werden, daß ein einfacher Satz geeigneter Regeln (Naturgesetze), wiederholt angewendet auf bestimmte Grundelemente (Moleküle), mit fast zwingender Notwendigkeit Strukturen beliebiger Komplexität erzeugt.
Aufgrund dieser Erkenntnisse sprechen manche Forscher von einer "Selbstorganisationsfähigkeit der Materie", eine Theorie, die zu weitreichenden Konsequenzen führt. Wie ist zum Beispiel das Leben auf unserem Planeten entstanden – ein einmaliger Zufall? Wenn die Materie unter der Einwirkung der Naturgesetze die Fähigkeit erlangt, sich selbst zu strukturieren, so ist vielleicht die individuelle Ausprägung der Lebensformen zufallsbedingt, nicht aber die Tatsache des organischen Lebens an sich – es mußte mit hoher Wahrscheinlichkeit entstehen, sobald die entsprechenden Rahmenbedingungen gegeben waren!

Am Anfang war der Punkt...

Wie aus einfachsten Voraussetzungen außerordentlich komplexe Muster entstehen können, soll nun ein kleines Software-Experiment auf dem CPC demonstrieren, ein wahres Ka-



binettstück der mathematischen Experimentalkunst. Eine derart kunstvolle Filigrangrafik, wie sie das Programm "Reproduktionsmuster" erzeugt, haben Sie wahrscheinlich auf Ihrem Bildschirm noch nicht gesehen!
Das Programm operiert mit den denkbar einfachsten Grundelementen, nämlich mit Bildpunkten, die entweder gesetzt (hell) oder zurückgesetzt (dunkel) sind. Dazu kommen noch zwei "Naturgesetze", die iterativ (wiederholt) auf den gesamten Bildschirm angewendet werden:

- Setze simultan alle Punkte, in deren Nachbarschaft (links, rechts, oben und unten) sich **genau ein** heller Punkt befindet.

Alle Punkte, die auf diese Weise bei einem Durchgang neu entstehen, werden als eine "Generation" bezeichnet. Mit Hilfe dieser Definition kann nun die zweite Regel formuliert werden:

- Nachdem die Generation (n) erzeugt wurde, lösche alle Punkte der Generation (n-2).

Die Punkte gehen also nach einiger Zeit an "Altersschwäche" zugrunde; nach jedem Durchgang sind exakt die jeweils letzten zwei Generationen auf dem Bildschirm zu sehen.
Solange die Regeln auf einen leeren Bildschirm angewendet werden, passiert natürlich überhaupt nichts – doch ein einziger Punkt, der in Zeile 410 des Programms als Kristallisationskeim gesetzt wird, bringt die Lawine ins Rollen. Bild 1 verdeutlicht das Prinzip und zeigt die Entwicklung der ersten fünf Generationen.
Das durch CALL &A000 aufgerufene Maschinenprogramm führt die Regeln aus und sorgt für die Bildschirmdarstellung. Übrigens wurden bei der Programmierung konsequent die in der Serie "Der gläserne CPC" (Heft 3 u. 4/86) dargestellten Techniken zur Erzeugung schneller Grafik benutzt. Die normalen Systemroutinen oder ein Basicprogramm wären hoffnungslos zu langsam, da pro Generation bis zu 384 000 mal Punkte auf ihre Farbe getestet werden müssen!
Die Darstellung der Generationen durch verschiedene Farben hat aber nicht nur ästhetische Gründe. Sie ermöglicht es dem Programm, die Generation (n-2) zu erkennen, ohne eine Unmenge von Koordinaten abzuspeichern. Weiterhin erlaubt die Farbdarstellung die quasi-synchrone Anwendung der Gesetze: Die während der Erzeugung einer Generation (n) bereits gesetzten Punkte werden noch nicht als Nachbarn gezählt. Natürlich kann der Computer nicht wie gefordert alle Bildpunkte gleichzeitig erfassen, sondern nur nacheinander behandeln – eine Tatsache, die durch geeignete Programmiertricks ausgeglichen werden muß, um Störungen zu vermeiden.
Wer noch gerne etwas mit diesem System experimentieren möchte, kann einmal eine veränderte Anfangssituation ausprobieren. Der in Zeile 410 erzeugte Punkt kann z.B. durch einen Buchstaben oder ein Grafiksymbol ersetzt werden. Sehr reizvoll sind auch Bilder, die aus mehreren auf dem Bildschirm verteilten Keimen entstehen. Obwohl die typische Charakteristik der erzeugten Grafiken erhalten bleibt, ergibt sich je nach Startfigur ein anderer Ablauf – und die Analogie zur Entstehung von Schneekristallen wird deutlich: Die Ausprägung der Form wird durch den Kristallkeim bestimmt (z.B. ein Staubkorn), die charakteristische Symmetrie jedoch durch die Gesetzmäßigkeiten der molekularen Bindungskräfte.

Der Zellautomat

Bevor wir uns einem weiteren, auf ähnlichen Grundlagen beruhenden Software-Experiment zuwenden, soll zunächst daran erinnert werden, auf welche Weise eigentlich der CPC Informationen verarbeitet.
Wie Sie sicherlich wissen, enthält Ihr Computer einen Mikroprozessor als zentrale Steuerungseinheit. Er liest binäre Werte aus den Speicherzellen, verknüpft sie in seinen internen Registern nach bestimmten Regeln, und legt das Ergebnis wieder in anderen Speicherzellen ab.
Diese Art der Informationsverarbeitung nennt man sequentiell: Kom-

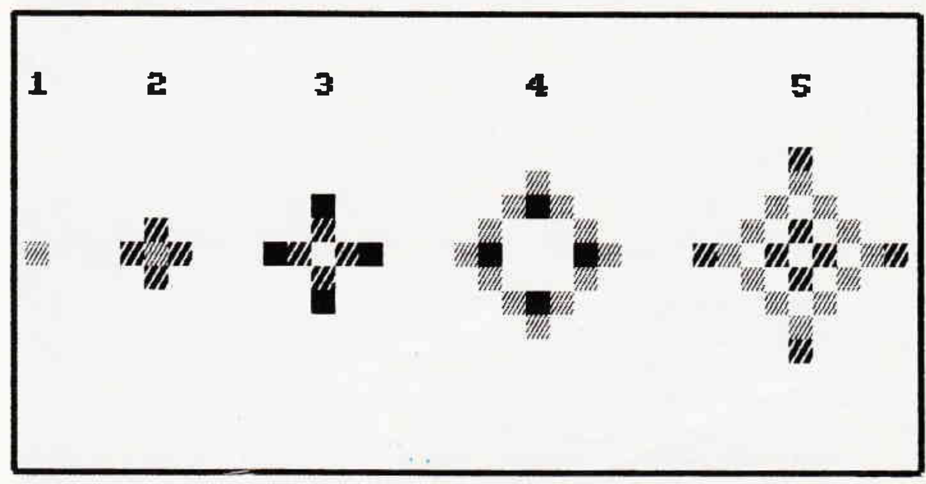


Bild 1: Reproduktionsmuster: Die ersten fünf Generationen

plizierte Berechnungen werden ausgeführt, indem die dazu notwendigen Schritte **nacheinander** Stück für Stück programmgesteuert abgearbeitet werden. Nach diesem Prinzip funktionieren praktisch alle Digitalcomputer, und nur die hohe Verarbeitungsgeschwindigkeit erzeugt mitunter den Eindruck, der Rechner könne mehrere Dinge gleichzeitig erledigen.

Der Computer ist damit ein getreues Abbild des sprachorientierten menschlichen Verstandes. Auch die verbale Informationsverarbeitung kann nur sequentiell erfolgen. Versuchen Sie einmal, bewußt zwei verschiedene Gedanken gleichzeitig zu denken – Sie werden sehen, es ist praktisch unmöglich.

Allerdings stellt der Verstand auch nur einen sehr kleinen Teilausschnitt des informatischen Systems "Mensch" dar. Schon die verschiedenen Formen der Sinneswahrnehmung erfordern ein Verarbeitungssystem, das nach einem ganz anderen Prinzip arbeitet. Es ist ja ohne weiteres möglich, mehrere Dinge gleichzeitig zu sehen oder zu fühlen – mit anderen Worten, der Mensch kann eine große Anzahl verschiedener Daten **parallel** erfassen.

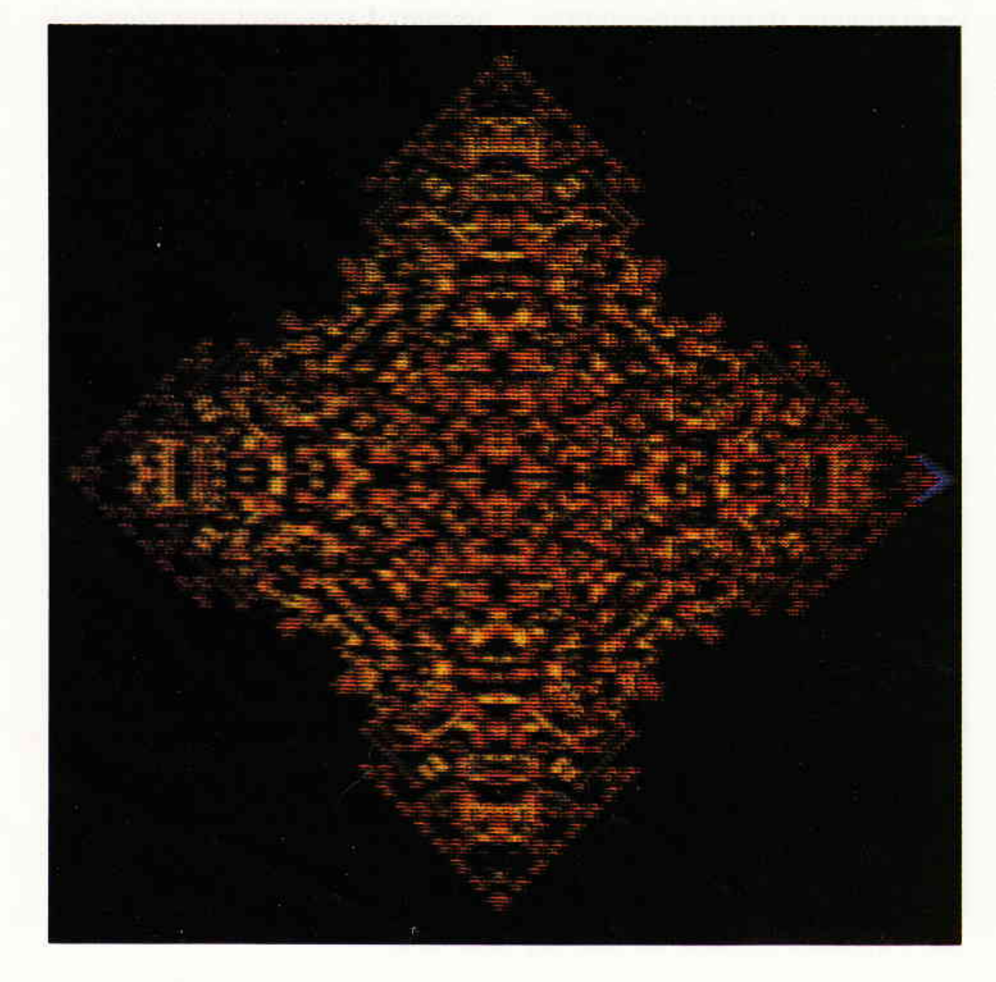
Der Verstand ist zwar in der Lage, die von den Sinnesorganen gelieferten Informationen auf einer abstrakten Ebene zu analysieren, doch für die unmittelbare Auswertung ist er effektiv zu langsam. Stellen Sie sich doch einmal vor, Sie müßten, um ein Haus zu erkennen, das Bild bewußt in ein Punktraster zerlegen, jedem Punkt einen Farb- und einen Helligkeitswert zuordnen, die Daten dann mit einem bereits gespeicherten Raster vergleichen und daraus einen Wert ermitteln, der Ihnen verrät, mit welcher Wahrscheinlichkeit es sich um ein Haus handelt... des langen Satzes kurzer Sinn: Sie wären praktisch handlungsunfähig!

Dieses Beispiel macht deutlich, warum selbst die besten und schnellsten Computer trotz aller Bemühungen der Forscher nicht in der Lage sind, im menschlichen Sinne zu sehen oder zu hören: Ihr Arbeitsprinzip ist dazu einfach ungeeignet. Zwar werden inzwischen schon Versuche mit Multiprozessor-Systemen unternommen, doch auch hierbei ergeben sich erhebliche praktische Schwierigkeiten. So kann man z.B. nicht erwarten, daß ein Rechner mit 100 Prozessoren 100 mal so schnell wie ein normaler Computer läuft, da ein großer Teil der Kapazität gebraucht wird, um den Datentransfer zwischen den verschiedenen Einheiten zu koordinieren.

Interessanterweise existiert jedoch ein Ansatz zur Lösung dieser Probleme, der bereits in der Vorcomputerzeit von dem Mathematiker John v. Neumann theoretisch untersucht wurde. Die schon 1936 von Alan M. Turing aufgeworfene Frage nach der möglichen Existenz einer universellen Denkmaschine veranlaßte ihn zur gedanklichen Konstruktion eines sogenannten "Zellautomaten".

Dabei handelt es sich um einen homogenen, zellulär unterteilten Raum, in dem jede Zelle eine bestimmte Anzahl von Zuständen annehmen kann (z.B. verschiedene Farben). Auf alle Zellen wird synchron ein Satz von Umwandlungsregeln (Transition Rules) angewendet, deren Ausführung im einzelnen von dem Zustand der jeweiligen Zelle und der angrenzenden Nachbarzellen abhängt. John v. Neumann konnte theoretisch nachweisen, daß eine Konfiguration aus 200.000 Zellen mit jeweils vier definierten Nachbarn und 29 Zustandsmöglichkeiten ausreicht, um im Prinzip jede denkbare Rechenoperation auszuführen.

Solch ein Zellautomat enthält also keinen zentralen Prozessor, sondern die Speicherzellen kommunizieren ohne Umschweife direkt miteinander. Mit jedem Taktimpuls, der im gesamten Raum eine Umwandlung laut Regelkatalog bewirkt, werden eine große Anzahl von Operationen gleichzeitig ausgeführt. Informationen erscheinen dabei nicht mehr als Bits und Bytes, sondern als Raum- oder Flächenmuster, die sich überlagern und gegenseitig beeinflussen können – ein Prinzip, das die Natur auch bei der Konstruktion des menschlichen Gehirns angewendet hat.



Das Spiel des Lebens

Die wohl bekannteste Veranschaulichung des Zellautomaten-Prinzips ist das 1970 vom britischen Mathematiker Conway erfundene "Game of Life". Wie in unserem vorherigen Experiment kann in diesem Spiel jede Zelle zwei verschiedene Zustände annehmen; sie ist also entweder "bewohnt" oder "unbewohnt". Allerdings wurde die Nachbarschaft auch auf die diagonal angrenzenden Zellen erweitert – jede Zelle hat also acht Nachbarzellen – und die Umwandlungsregeln wurden so formuliert, daß die Abläufe recht realistisch die Folgen von Isolation, Überbevölkerung oder Kooperation in einer Gesellschaft lebendiger Wesen darstellen. Sie lassen sich durch zwei einfache Kommandos erfassen:

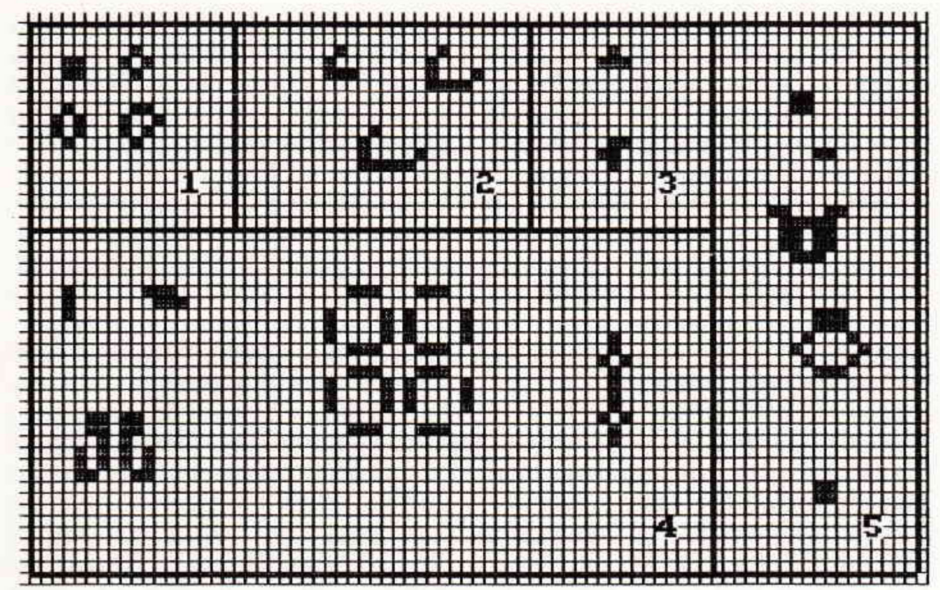


Bild 2: Einige GAME OF LIFE-Konfigurationen

- Besetze synchron alle leeren Zellen, die **genau drei** Nachbarn haben.
- Lösche gleichzeitig (!) alle Zellen, die weniger als zwei oder mehr als drei Nachbarn haben.

Eine weitere Darstellungsmöglichkeit für diese Gesetze ist ein computergerecht gestalteter Regelkatalog in Tabellenform. Er gibt die Belegung einer Zelle in der Generation (n+1) in Abhängigkeit von ihrem momentanen Zustand und der Anzahl der Nachbarn an. Eine 1 bedeutet, daß die Zelle in der nächsten Generation bewohnt sein wird, eine 0 steht dagegen für eine leere Zelle:

	Anzahl der Nachbarn								
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Zellen- 0	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**0**
zustand 1	**0**	**0**	**1**	**1**	**0**	**0**	**0**	**0**	**0**

Auf den ersten Blick scheinen die Lebensbedingungen der Life-Einzeller also ziemlich schlecht zu sein – sie reagieren sehr empfindlich auf Einsamkeit oder Überbevölkerung. Einige Versuche zeigen jedoch recht schnell, daß gerade diese Einschränkungen dem System seine Dynamik verleihen und zu hochinteressanten Abläufen führen. Nicht umsonst hat dieses Spiel nach seiner Entstehung an Universitäten und wissenschaftlichen Instituten für allerhand Aufsehen und mitunter sogar Ärger gesorgt, da die Terminals der Rechenanlagen regelmäßig von Life-Fanatikern blockiert wurden – für jedermann verfügbare Heimcomputer gab es ja in den siebziger Jahren noch nicht.

Doch die Zeiten haben sich glücklicherweise geändert, und Sie können mit Hilfe des Programms "GAME OF LIFE" stundenlang auf Ihrem CPC experimentieren, ohne befürchten zu müssen, daß Sie im nächsten Moment das Betriebssystem wegen Überschreitung der Rechenzeit aus dem Rechner wirft.

In dem Programm werden die Zellen nicht wie bei den Reproduktionsmustern pixelweise, sondern etwas vergrößert dargestellt; insgesamt steht eine Fläche mit 50*80 = 4000 Zellen zur Verfügung. Die Umwandlungsregeln führt wieder ein kleines Maschinenprogramm aus, es schafft etwa zwei Generationen pro Sekunde. Besonderer Wert wurde auf eine extrem schnelle Bildschirmausgabe gelegt – in der Tat scheinen die Umwandlungen auf der gesamten Fläche gleichzeitig zu erfolgen.

Nach dem Programmstart erscheint zunächst das leere Spielfeld, das mit einem Karomuster vorstrukturiert ist, um eine genaue Analyse der einzelnen Abläufe zu ermöglichen. Um sich einen Eindruck von dem Treiben der Life-Einzeller zu verschaffen, können Sie zunächst die Taste <Z> drücken, worauf der Rechner eine Zufallsverteilung auf dem Bildschirm produziert. Warten Sie eine Weile, bis die Fläche einigermaßen voll ist, und brechen Sie dann den Vorgang mit <ENTER> ab. Mit Hilfe der <SPACE>-Taste können Sie nun die jeweils nächste Generation erzeugen, oder auch mit <A> die Generationswechsel automatisch ablaufen lassen. Auch diese Funktion wird mit <ENTER> abgebrochen.

Um gezielt die Eigenschaften bestimmter Konfigurationen zu untersuchen, kann mit <E> ein kleiner Editor angewählt werden. Der Cursor wird wie üblich mit den Pfeiltasten gesteuert; die Taste <1> belegt die Zelle auf Cursorposition, die Taste <0> löscht sie entsprechend. Um ganze Reihen von von Life-Wesen zu produzieren, können Sie auch die Taste <1> gedrückt halten und den Cursor über den Bildschirm wandern lassen, er hinterläßt dann eine Spur von gefüllten Zellen. Mit <ENTER> wird Edit-Modus wieder verlassen und mit <L> bei Bedarf die gesamte Fläche gelöscht.

Bild 2 zeigt einige typische Life-Konfigurationen. Die mit 1 bezifferte Gruppe enthält vier stabile Figuren, die bei den Generationswechseln unverändert bleiben. Gruppe 2 zeigt drei sogenannte "Gleiter". Bei diesen

Figuren handelt es sich um Oszillatoren, die periodisch ihre alte Form wiedererlangen, sich dabei aber um ein kleines Stück weiterbewegen. Stationäre Oszillatoren mit einer Periode von 2 bis 15 Generationen finden Sie dagegen in der Gruppe 4 versammelt. Ein besonders eigenartiges Verhalten zeigt der 30-Generationen-Oszillator, der die Gruppierung 5 bildet: Er produziert, ohne dabei an Substanz zu verlieren, eine Serie kleiner Gleiter, die nach rechts abwandern.

Die Gruppe 3 enthält noch zwei Figuren, die ein expansives Verhalten zeigen. Während die obere Figur jedoch schon nach wenigen Generationen in vier stationäre Oszillatoren zerfällt, ist die untere Figur ohne weiteres in der Lage, den gesamten Bildschirm mit ihren Nachkommen zu füllen. Sie gilt als die kleinste Figur, die sich beliebig weit ausdehnt; bisher konnte noch kein definierter Endzustand ermittelt werden.

Wenn Sie die abgebildeten Muster alle gemeinsam testen wollen, entfernen Sie am besten zunächst diese expansive Figur und die drei Gleiter aus dem Sortiment, um die anderen Figuren nicht zu stören. Ansonsten ist es aber sehr interessant, was sich bei einer Kollision verschiedener Muster abspielt: Sowohl eine totale Auslöschung als auch die Entstehung neuer Figuren kann die Folge sein. Durch die eifrigen Bemühungen der vielen Life-Forscher ist es inzwischen sogar gelungen, Muster zu finden, die durch gegenseitige Beeinflussung logische Funktionen wie etwa NOT, AND, OR oder XOR simulieren – ein wichtiger Schritt in Richtung eines praktisch anwendbaren Zellautomaten.

Ein Problem, das die Realisierung eines auch nur kleinen Zellrechners auf dem CPC verhindert, ist leider die unvermeidliche Spielfeldbegrenzung – selbst eine 320*200-Fläche wäre noch viel zu klein. Das Programm betrachtet alle Zellen außerhalb der Fläche als leer, und daraus ergeben sich empfindliche Störungen, sobald ein Muster an den Rand stößt. Auf interessante Weise arrangiert sich allerdings der Gleiter oben rechts mit dieser Beschränkung: Er wird reflektiert und verwandelt sich dabei in einen kleineren Gleiter!

Wie Sie bei eigenen Experimenten feststellen können, führt die Begrenzung des Lebensraumes dazu, daß alle Konfigurationen nach einiger Zeit steckenbleiben; es bleiben nur noch stabile Muster und Oszillatoren übrig, die keine grundsätzlich neuen Entwicklungen mehr bewirken können. Doch dieses Phänomen hat noch eine andere Ursache: Im Game of Life ist der "schöpferische Zufall" nicht vorgesehen. Der gesamte Prozeß läuft deterministisch ab, d.h. mit einer gegebenen Anfangssituation steht praktisch schon in allen Einzelheiten fest, was passieren wird. Sie können aber einmal erforschen, was die folgenden zusätzlichen Programmzeilen bewirken:

```
981 f=1
982 re=INT(RND*50)+1
982 sp=INT(RND*80)+1
983 GOSUB 1060
```

Wenn Sie jetzt den automatischen Generationswechsel mit <A> anwählen, wird bei jedem Durchgang ein neues Wesen an einem zufälligen Ort geboren und bringt etwas Abwechslung in die Angelegenheit – wie bereits in der vorigen Folge angedeutet, spielt der Zufall bei der Erzeugung dynamischer ("lebendiger") Strukturen ja eine wichtige Rolle.

Und es gibt noch eine weitere sehr interessante Experimentiermöglichkeit: In den DATA-Zeilen 530 und 540 steht der vorhin bereits erwähnte Regelkatalog für das Game of Life. Hier können Sie natürlich die Nullen und Einsen auch anders verteilen und damit neue Umwandlungsgesetze gestalten. Die Abläufe, die sich aus einem veränderten Regelkatalog ergeben, stellen ein recht unerforschtes Gebiet dar und bieten noch viel Raum für aufregende Entdeckungen.

Evolution im Computer?

Nach dieser schon recht eindrucksvollen Simulation eines Zellautomaten drängt sich die Frage auf: Wieso ist eigentlich eine solche Maschine noch nicht gebaut worden? An der Hardware dürfte es nicht scheitern, die nötigen Chips wären mit dem heutigen technischen Stand sicherlich realisierbar. Doch das Problem liegt ganz woanders: Kein Mensch weiß, wie man einen Zellautomaten programmieren könnte.

Wie bereits erörtert wurde, beruht jede Sprache – und damit auch jede Programmiersprache – auf einer sequentiellen Logik. Zwar können wir auch über Ereignisse sprechen, die simultan stattfinden, jedoch nur, indem wir sie **nacheinander** beschreiben. Deshalb ist es sehr zweifelhaft, ob ein leistungsfähiger Zellautomat, der hunderte und tausende von Operationen gleichzeitig durchführt und dabei unzählige sich kreuz und quer gegenseitig beeinflussende Muster bildet, jemals mit den Mitteln eines sequentiell arbeitenden Verstandes beherrscht werden kann. Bisher ist nur ein einziger Ausweg aus diesem Dilemma sichtbar: Die Maschine muß sich selbst programmieren!

Wenn Sie Ihren CPC programmieren, so formulieren Sie mit Hilfe einer Programmiersprache einen Algorithmus, der Schritt für Schritt angibt, wie aus einer Situation A das Resultat B erzeugt wird. Wie einfach wäre es doch, wenn Sie den Computer einschalten und zu ihm sprechen könnten: „Hier, mein elektronischer Freund, das ist ein Schachbrett, das sind die Figuren, und so gehen die Regeln... und du hast gewonnen, wenn du meinen König eroberst, bevor ich deinen erwische. Spiele so, daß du nach Möglichkeit gewinnst!"

Sehen Sie – genau das müßte der Zellautomat können: selbständig Lösungswege finden. Wie bereits gezeigt wurde, stellt es kein Problem dar, sich selbst organisierende Systeme zu erschaffen – doch mit ein paar Umwandlungsregeln, die irgendwelche Muster erzeugen, ist es in diesem Fall nicht getan. Um gut Schach zu spielen, müßte die Maschine **zielgerichtet** problembezogene Strukturen produzieren, ungünstige Muster verwerfen, Erfahrungen sammeln und selektiv auswerten... mit anderen Worten, einen Lernvorgang, eine Evolution nachvollziehen.

Doch die Formulierung von Bewertungsschemata, die einen sich selbst organisierenden Prozeß zielgerichtet ablaufen lassen, ist bisher ein theoretisch nur ansatzweise gelöstes Problem. Interessante Hinweise liefert allerdings ein Forschungsbereich, der sich mit dem Grenzgebiet zwischen Biologie und Informatik beschäftigt: Welche Strategien hat die Natur bei der Entwicklung des Lebens angewendet? Welche mathematisch-statistischen Gesetzmäßigkeiten tragen dazu bei, daß sich bestimmte Arten entfalten, während andere aussterben?

Die nächste Folge des Software-Experiments wird noch etwas Material zu diesem Thema liefern. Wer sich intensiver mit diesem Gebiet beschäftigen möchte, dem sei das Buch "Das Spiel – Naturgesetze steuern den Zufall" von Manfred Eigen/Ruthild Winkler (Piper Verlag) empfohlen. Es werden eine ganze Reihe weiterer Simulationsspiele vorgestellt, die sich auch auf einem Heimcomputer realisieren lassen – eine gute Grundlage für alle Computer-Forscher und solche, die es werden wollen!

(M. Uphoff)

für 464-664-6128

```
10 '******* Reproduktionsmuster *******          [2045]
20 '                                              [117]
30 MEMORY &9FFF                                   [134]
40 FOR adr=&A000 TO &A0AB                         [872]
50 READ a$:v=VAL("&"+a$)                          [870]
60 s=s+v:POKE adr,v                               [1258]
70 NEXT adr                                       [547]
80 IF s<>19012 THEN PRINT"DATAFEHLER !":EN        [3572]
D
90 '                                              [117]
100 DATA EF,03,A0,21,A9,A0,E5,DD                  [948]
110 DATA E1,7E,23,46,23,4E,77,2B                  [1223]
120 DATA 71,2B,70,16,88,21,00,C0                  [1455]
130 DATA 0E,C8,06,50,E5,7E,A2,20                  [965]
140 DATA 4D,5F,7E,CB,02,30,08,3E                  [1574]
150 DATA 50,B8,28,06,2B,7E,23,CD                  [1029]
160 DATA A0,A0,CB,0A,7E,CB,0A,30                  [1593]
170 DATA 08,3E,01,B8,28,06,23,7E                  [1220]
180 DATA 2B,CD,A0,A0,CB,02,3E,C8                  [1487]
190 DATA B9,28,09,E5,CD,2D,0C,7E                  [1825]
200 DATA E1,CD,A0,A0,3E,01,B9,28                  [907]
210 DATA 09,E5,CD,13,0C,7E,E1,CD                  [866]
220 DATA A0,A0,3E,01,BB,20,07,3A                  [924]
230 DATA A9,A0,AE,A2,AE,77,CB,0A                  [1856]
240 DATA 30,AB,23,10,A8,E1,CD,13                  [807]
250 DATA 0C,0D,20,9E,3A,AA,A0,5F                  [1174]
260 DATA 21,00,C0,0E,C8,06,50,E5                  [2209]
270 DATA 7E,AB,A2,20,04,AE,A2,AE                  [2061]
280 DATA 77,CB,0A,30,F3,23,10,F0                  [1496]
290 DATA E1,CD,13,0C,0D,20,E6,C9                  [846]
300 DATA A2,C8,DD,AE,00,A2,C8,1C                  [1859]
310 DATA C9,0F,F0,FF                              [911]
320 '                                             [117]
330 REM Pokes fuer CPC 664                        [967]
340 'POKE &A04D,&35:POKE &A05B,&1B:POKE &A        [3665]
077,&1B:POKE &A09A,&1B
350 '                                             [117]
360 REM Pokes fuer CPC 6128                       [1071]
370 'POKE &A04D,&39:POKE &A05B,&1F:POKE &A        [4409]
077,&1F:POKE &A09A,&1F
380 '                                             [117]
390 MODE 1:BORDER 0                               [1436]
400 INK 0,0:INK 1,6:INK 2,15:INK 3,2              [796]
410 PLOT 320,200,2                                [619]
420 WHILE INKEY$=""                               [1095]
430 CALL &A000                                    [637]
440 WEND                                          [390]
```

```
10 '********** GAME OF LIFE **********            [2426]
20 '                                              [117]
30 '********** Basiclader fuer M-Code             [2932]
40 MEMORY &7FFF                                   [150]
50 FOR adr=&8000 TO &8097                         [865]
60 READ a$:v=VAL("&"+a$)                          [870]
70 s=s+v:POKE adr,v                               [1258]
80 NEXT adr                                       [547]
90 IF s<>16737 THEN PRINT"DATAFEHLER !":EN        [1704]
D
100 '                                             [117]
110 DATA DD,21,EA,80,11,92,91,26                  [1426]
120 DATA A2,0E,32,06,50,DD,23,13                  [1133]
130 DATA DD,7E,AD,DD,86,AE,DD,86                  [1755]
140 DATA AF,DD,86,FF,DD,86,01,DD                  [537]
150 DATA 86,51,DD,86,52,DD,86,53                  [1413]
160 DATA DD,CB,00,46,28,02,C6,09                  [1258]
170 DATA 6F,6E,7D,DD,AE,00,0F,B5                  [1391]
180 DATA 12,DD,23,13,10,D2,DD,23                  [947]
190 DATA 13,0D,20,C7,21,92,91,11                  [1167]
200 DATA EA,80,01,04,10,ED,B0,21                  [1072]
210 DATA 00,E0,E5,26,C0,11,EA,80                  [726]
220 DATA 0E,32,06,50,13,1A,CB,7F                  [1468]
230 DATA 28,15,E6,01,12,D5,E5,11                  [588]
240 DATA 00,08,3E,0E,28,02,3E,E0                  [733]
250 DATA 77,19,77,19,77,E1,D1,23                  [1460]
260 DATA 13,10,E2,E3,13,0D,20,DA                  [1339]
270 DATA E1,C9,3E,80,CD,8B,80,CD                  [2068]
280 DATA 4F,80,AF,21,98,80,54,5D                  [1019]
290 DATA 13,01,4F,21,77,ED,B0,C9                  [1109]
300 '                                             [117]
310 '********** Initialsierung                    [1671]
320 MODE 1                                        [506]
330 DEFINT a-z                                    [553]
340 INK 0,0:INK 1,13:INK 2,3:INK 3,24             [1313]
350 BORDER 0                                      [1008]
360 PLOT -2,-2,3                                  [801]
370 SYMBOL 250,&F8,&88,&88,&88,&F8,0,0,0          [2234]
380 cu$=CHR$(250)                                 [294]
390 xon$=CHR$(23)+CHR$(1)                         [1869]
400 xoff$=CHR$(23)+CHR$(0)                        [1386]
410 feld=&8098:clrfeld=&8082:gen=&8000            [2435]
420 t1$="EZLA "                                   [255]
430 t2$=CHR$(13)                                  [475]
440 FOR c=240 TO 243                              [1167]
450 t2$=t2$+CHR$(c)                               [760]
460 NEXT                                          [350]
470 CALL clrfeld:REM Feld loeschen                [1773]
480 '                                             [117]
490 '********** Regelkatalog einlesen             [2607]
500 FOR adr=&A200 TO &A211                        [822]
510 READ et:POKE adr,et                           [1295]
520 NEXT                                          [350]
530 DATA 0,0,0,1,0,0,0,0,0                        [1344]
540 DATA 0,0,1,1,0,0,0,0,0                        [1100]
550 '                                             [117]
560 '********** Tastaturabfrage                   [2542]
570 'E = Editor                                   [493]
580 'Z = Zufallsverteilung                        [1610]
590 'L = Spielfeld loeschen                       [2043]
600 'A = autom. Generationswechsel                [3382]
610 '<SPACE> = naechste Gen. berechnen            [2689]
620 '<ENTER> = Funktionen E,Z,A beenden           [1247]
630 '                                             [117]
640 a$=UPPER$(INKEY$):IF a$="" THEN 640           [1916]
650 ON INSTR(t1$,a$) GOTO 690,850,940,970,        [3579]
1030
660 GOTO 640                                      [373]
670 '                                             [117]
680 '********** Editor                            [680]
690 PRINT xon$;                                   [1057]
700 re=1:sp=1                                     [854]
710 TAG:MOVE 8*sp-10,408-8*re:PRINT cu$;          [2214]
720 a$=INKEY$:IF a$="" THEN 720                   [1209]
730 MOVER -16,0:PRINT cu$;:TAGOFF                 [1681]
740 IF NOT INKEY(64) THEN f=1:GOSUB 1060          [891]
750 IF NOT INKEY(32) THEN f=0:GOSUB 1060          [1230]
760 ON INSTR(t2$,a$) GOTO 640,790,800,810,        [2116]
820
770 GOTO 710                                      [431]
780 '                                             [117]
790 IF re>1 THEN re=re-1:GOTO 710 ELSE 710        [1826]
800 IF re<50 THEN re=re+1:GOTO 710 ELSE 71        [1456]
0
810 IF sp>1 THEN sp=sp-1:GOTO 710 ELSE 710        [1068]
820 IF sp<80 THEN sp=sp+1:GOTO 710 ELSE 71        [1669]
0
830 '                                             [117]
840 '********** Zufallsverteilung                 [1177]
850 f=1                                           [395]
860 WHILE INKEY$<>CHR$(13)                        [1261]
870 re=INT(RND*50)+1                              [574]
880 sp=INT(RND*80)+1                              [1196]
890 GOSUB 1060                                    [823]
900 WEND                                          [390]
910 GOTO 640                                      [373]
920 '                                             [117]
930 '********** Spielfeld loeschen                [2036]
940 CALL clrfeld:GOTO 640                         [1188]
950 '                                             [117]
960 '********* Autom. Generationswechel           [2073]
970 WHILE INKEY$<>CHR$(13)                        [1261]
980 CALL gen                                      [257]
990 WEND                                          [390]
1000 GOTO 640                                     [373]
1010 '                                            [117]
1020 '********* Naechste Gen. berechnen           [2147]
1030 CALL gen:GOTO 640                            [1148]
1040 '                                            [117]
1050 '****** SUB Zelle belegen/loeschen           [2027]
1060 POKE feld+82*re+sp,f                         [1610]
1070 PRINT xoff$;                                 [1313]
1080 IF f=1 THEN PLOT -2,-2,1 ELSE PLOT -2        [2593]
,-2,2
1090 FOR x=8*sp-8 TO 8*sp-4 STEP 2                [2583]
1100 FOR y=402-8*re TO 406-8*re STEP 2            [2874]
1110 PLOT x,y                                     [471]
1120 NEXT y,x                                     [397]
1130 PRINT xon$;:PLOT -2,-2,3                     [1947]
1140 RETURN                                       [555]
```

Backnang

Computer-Fans finden bei uns alles von:



Berlin

Ihr starker Schneider-Partner



Schöneberger Straße 5 • 1000 Berlin 42 • Tel. 030-752 91 50/60



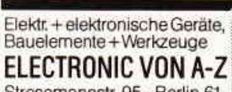
A+C Vertrieb, 1 Berlin 44, Emser Str. 18

Ihre COMPUTEREI

Schneider COMPUTER DIVISION

Hardware
Software
Beratung
Literatur

Karl-Marx-Str. 243
1000 Berlin 44
Am U-Bhf. Neukölln
030/6841098





Bonn

RADIO-FERNSEHEN
HIFI-VIDEO
Schäfer
SERVICE SERVICE SERVICE SERVICE

Plittersdorfer Straße 206 Telefon 36 40 29

Castrop-Rauxel



Düsseldorf

...alles für die Schneider-Computer!







Freiburg



Hamburg

Jetzt auch bei uns:
Joyce und CPC 464 + 6128

Anwenderprogramme z.B. für Joyce: Wordstar 3.0, d Base II, Multiplan, Finanzbuchhaltung, Business Pack, DR DRAW, DR GRAPH, Schach

Große Auswahl an Spiel- und Anwenderprogrammen, Zubehör und Literatur für SCHNEIDER und COMMODORE.

Gärtnerstr. 5 · 2000 Hamburg 20 · Tel. 420 48 21



autorisierter Fachhändler

HAMBURGER SOFT WARE LADEN

Löhne/Ostwestfalen



Schneider Vertragshändler & Servicecenter
Hard- & Software von A–Z für Ostwestfalen
FRITZ OBERMEIER
•Computer•HiFi•Video•TV•
alles für Schneider vom 464 – Joyce
am Hauptbahnhof • Bünder Str. 20 • 4972 Löhne 1 • Tel. 05732/3246

Nürnberg



Hochstraße 11
8500 Nürnberg 80
Tel. 0911/28 90 28

Schneider COMPUTER DIVISION

Wir führen zu den original SCHNEIDER-Produkten Software, Bücher und Zubehör verschiedener Firmen wie DATA BECKER, VORTEX, CUMANA, ISS, RUSHWARE, MARKT & TECHNIK, SYBEX, VOGEL-Verlag usw.!



Rüsselsheim

commodore Schneider-Computer
Dipl. Ing. Neuderth
Frankfurter Str. 23/Friedensplatz, 6090 Rüsselsheim
Tel. (0 61 42) 6 84 55, Tx. vide d 4 182 982

Würzburg

HALLER GmbH, Würzburg
der Spezialist für alle Schneider-Computer.
Hardware – Software – Zubehör – Service
Büttner Str. 29 (hinter Kaufhaus Hertie)
Telefon: 0931/16705

Achtung! An alle CPC-User-Clubs!
Suchen Sie noch Mitglieder oder wollen Kontakte zu anderen CPC-Usern oder Clubs knüpfen? Dann schreiben Sie uns doch einfach. Wir veröffentlichen jede Anschrift eines CPC-User-Cubs gratis! Falls auch Ihr User-Club sich einmal vorstellen möchte (evtl. mit Foto), nutzen Sie unsere Zeitschrift als Verbindungsglied und Kontaktadresse!

CPC-Nutzerclub "Fehler im System"
sucht weitere Mitglieder aus Ostwestfalen & Südniedersachsen
Näheres bei:
Fritz-Peter Nonnenbruch,
Splittenbrede 11,
4800 Bielefeld 1,
Telefon: (05 21) 88 79 70
oder
Gerd Engelbarth,
Aschener Weg 1,
4509 Dissen,
Telefon: (0 54 21) 51 83

IAC – International Amstrad Club
Kölner Straße 66,
5630 Remscheid 11,
Telefon: (0 21 91) 6 51 21

Syntax Error
Computerclub Nordhorn
(alle Systeme)
Infos bei:
Fred Weber,
Laarstraße 23,
4460 Nordhorn
Telefon: (0 59 21) 1 31 49
oder:
Heinz Elbers,
Württemberger Straße 1,
4460 Nordhorn,
Telefon: (0 59 21) 7 54 78

Märkischer Kreis MK
"Ernsthafte" CPC 664/464 Anwender zum Erfahrungsaustausch und zur evtl. CPC-Clubgründung sucht:
Antonius Gusik DL1DK,
Bromberger Straße 31,
5880 Lüdenscheid,
Telefon: (0 23 51) 8 39 11

CPC-Club MAGIC KEY
sucht Mitglieder im Raum Bremen.
Infos bei:
Michael Hollmann,
Starnberger Straße 46,
2800 Bremen 1

ROMSOFT
sucht Kontakt zu anderen CPC-Clubs und -Usern.
Geschäftsführer:
Michael George,
Waldstraße 13,
6630 Saarlouis 5,
Telefon: (0 68 31) 67 01

A.N.W.U.C.
sucht Kontakt zu anderen CPC-Clubs
C. Heyliger (President),
41 Millwall Close,
Gorton, Manchester.
M18 8LL.

JOYCE USER-CLUB NÜRNBERG
Suche Kontakte zu einem Joyce-User-Club in Nürnberg bzw. Interessenten zur Gründung eines solchen.
Ansgar Zerfaß,
Ammanstraße 26,
8500 Nürnberg 40,
Telefon: (09 11) 43 67 78

Kontakt zu Schneider CPC 464-Besitzern im Raum Hamm gesucht.
Henrik Hünerhoff,
Caldenhofer Weg 44,
4700 Hamm 1

Schneider Computerclub Hamburg
sucht Mitglieder in ganz Europa.
Alexander Scharf,
Heimfelderstraße 60,
2100 Hamburg 90,
Telefon: (0 40) 7 90 93 53

CPC-User in Rüsselsheim
suchen noch andere CPC-User zwecks Erfahrungsaustausch.
Bei Interesse bitte melden bei:
Markus Jöst,
Amorbacher Straße 52,
6090 Rüsselsheim,
Telefon: (0 61 42) 3 33 31

Berliner User Club
sucht Mitglieder, auch weibliche, jeden Alters.
Infos von:
Wolfgang Windorpski,
Gritznerstraße 38,
1000 Berlin 41,
Telefon: (0 30) 8 22 77 50

CPC Clubgründung angestrebt
für Kreis Borken.
Interessierte bitte melden bei:
Dominique Weiß,
Südring 6,
4282 Velen – Ramsdorf

Schneider User Club Heilbronn
sucht noch Mitglieder aus dem Großraum Heilbronn.
Bitte melden bei:
Mirko Pecoroni,
Deutschherrenstraße 6,
7107 Bad Friedrichshall-4,
Telefon: (0 71 36/67 01

S.C.C.N-H
Schneider Computer Club Neunkirchen-Herdorf
Infos bei:
Oliver Thiele,
Am Kirchweg 49,
5908 Neunkirchen 5,
Telefon: (0 27 35) 31 66

JOYCE-Userclub Neugründung
Saarland
Kontakt und Info:
Arthur Borens,
Im Technologiepark Dillingen,
Postfach 1105,
Telefon: (0 68 31) 7 30 52,
D-6638 Dillingen/Saar

Schneider-Computerclub e.V.
Abt. Ostwestfalen/Lippe
Info: Computer-Freunde/Löhne e.V.
Postfach 1101
Telefon: 05732/3396
4972 Löhne 1

UCC-GREFRATH
sucht Kontakt zu anderen Clubs
Bitte melden bei:
Carsten Hoener,
Velourstraße 19,
4155 Grefrath 1

Schneider Computer Club
Heinsberg/Rheinland
Helmut Krings,
Josef-Gaspers-Straße 25,
5138 Heinsberg 1,
Telefon: (0 24 52) 57 10

Schneider Computer Club Hamburg sucht Mitglieder in ganz Europa.
Alexander Scharf,
Heimfelderstraße 60,
2100 Hamburg 90,
Telefon: (0 40) 7 90 93 53

Clubgründung angestrebt
CPC 464 Raum Engelskirchen
Siegfried Meyer,
Lindenpfuhlstraße 32,
5250 Engelskirchen

Biete an Software

★ ★ ★ **SCHNEIDER CPC 464** ★ ★ ★
★ **Hannover's** ★
★ **SOFTWARETHEK Nr. 1** ★
★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★
Mastertronic-Games a. C 9.90
Warlord/Centre Court a. C 29.—
Music Composer/Death Pit a. C 32.—
Fighting Warrior a. C 36.—
Hyper Sports C/D 42/54.—
The way of the Tiger C/D 36/52.—
Yie ar Kung Fu D 52.—
Colossus 4 Chess D 59.—
Hi Rise/The Devils Crown a. D 52.—
Textomat plus CPC 6128 D 198.—
GRAPHICS Lightpen+Software 98.—
sofort **CSJ NEWS** anfordern
★ ★ **CSJ COMPUTERSOFT JONIGK** ★ ★
★ **An der Tiefenriede 27, 3000 Hannover 1** ★
★ ★ ★ **Tel.: 0511/886383** ★ ★ ★
★ G ★

SOFTWARE – TRAUMPREISE
Weit über 1000 Titel lieferbar.
Alle gängigen Heimcomputer,
auch für Amiga 520ST, IBM+Comp.
Werden Sie Clubmitglied!
Bis zu 30 % Nachlaß!!!
ZODIAC-Softwareversand-Computerclub
Zollstr. 1, 2810 Verden, Tel.: 04231/2882
Händleranfragen erwünscht. G

Satellitenbahnberechnung für Funkamateure auf Joyce 8256. Alle Pr. laufen unter CP/M. Info g. Freiumschlag.
M. Maday, DC9ZP, Bismarckstr. 28, 2393 SÖRUP

SUPERCOPY – DISC-KOPIERPROGRAMM
100 % MC / 2. Laufwerk / alle Fremdformate / 43 Tracks / alle Sektorgrößen / Sekt. mit gel. DATA AD. M. jede mögl. Sektoranz. DM 79,– zzgl. VK, V-Scheck, NN; Th. Scholl, Tel.: 0711/362983 ab 17 Uhr
7300 Esslingen Wiflingshauserstr. 83 G

DIAS ordnen mit dem CPC
bis 25000 Dias pro Diskette.
Suchzeit ca. 1 Sekunde.
Info gegen Rückporto.
W. Grotkasten, Birnenweg 6,
7060 Schomdorf, Tel.: 07181/42846 G

SCHNEIDER-CPC-Programme
aus Eigenentwicklung. Gratisinfo anfordern bei Friedrich Neuper, Postfach 72, 8473 Pfreimd G

TASWORD-D+COPY+PRINT, D 3",
orig, neu, nur 120,–, Tel.: 0821/706502

★ MICA DM 176,—
KASSEMBLE DM 79,90
★ STAR WRITER I DM 169,—
MOS DM 169,—
★
ASS-WARE, Postfach 1441
★ 4830 Gütersloh, Tel. 05241/29953 G

■ **JOYCE** ■■ **LocoScript-Fibel** ■
Präzise, systematische Kurzanleitung.
19,– DM+Porto+ Nachnahme.
Walter Ehlers, 2000 Hamburg 56, Rheingoldweg 5, Tel.: 040/813721

Prof. Musikverwaltung! Tel. 05031/4302

SOFTWARE für Joyce
★ Adressverwaltung m. Briefdruck (variable Eingabe, Sortieren, Selektieren, komfortable Serienbriefe, usw.) DM 128,–
★ Biorhythmus (tab./graph.) DM 38,–
Vorkasse oder Nachnahme
G. Niemann, Karl-Valentin-Weg 10
8011 Kirchheim

Original Software für CPC 464
Datamat, Textomat, Budget-Manag.
m. Handbuch+Disk. je DM 80,–
zus. DM 200,–
☎ 0212/336718 ab 19.00 Uhr

Achtung! ★ ★ **Alles Originale** ★ ★
3 Zoll Elite dtsch. 59,– / Marco Polo 1 25,– / Budget Manag. 45,– / Thorr-Trilogie 45,–/Win. Sports 30,– DM.
★ 5,25 Zoll von Data M. Kassenbuch 20,– Verwaltungsarch Kassetten: Master Chess 30,– DM.
Das Geheimnis / Harrier Attack / Ghostbusters / Gremlins / Manic Miner / Codename Mat je 20,– DM
Alien Break In / Atom Smasher / Erbert / The Wild Bunch / Grand Prix Driver je 10,– DM
Chiller / Nonterraqueous / Locomotion / Finders Keepers je 7,– DM
Schneider Basic T. 1+2 59,– DM
dk'tronics Graphiks LPEN 40,– DM
NLQ 401 mit Traktor 499,– DM
dataphon s21d 150,– DM
Alle Artikel zuzügl. 3,– Porto+ Verpackung
Telefon: (040) 6 47 25 00 G

JOYCE PCW 8256
FAKTURA-CONTROL, 3"-Disk 89,– DM
Fakturieren mit Adress-Verw. und autom. Mahnprogr., offene Posten je Kunde.
FAKTURA-CONTROL PLUS, 3"-Disk 139,– DM
zusätzl. mit Artikel-Bestandsdatei, Umsatz- u. Mindestbestandskontrolle.
BAUFINANZIERUNG, 3"-Disk 189,– DM
Für Finanzierungsberater / Makler
Indiv. Programme zum Festpreis.
HASHAGEN, Tel.: 06073/61993
Eckstr. 11, 6113 Babenhausen 3,

Jetzt aber schnell
LOHNSTEUER-JAHRESAUSGLEICH '85
Bis zum 30.9. müssen Ihre Anträge beim Finanzamt sein! Für Schneider 664 und 6128 bieten wir das Rechenprogramm für Ihren Lohnsteuer- oder Einkommensteuer-Jahresausgleich:
PRIVAT – '85
Damit erhalten Sie die kompletten und aufgeschlüsselten Ausrechnungen. Einfacher geht es nicht! Für nur 60,00 DM plus Nachnahmegebühr. Den Prospekt gibt es für 80 Pfennig in Briefmarken. Auch für Commodore C 64, Atari 260/520 und Enterprise 128 geeignet.
dialog partner in berlin gmbh
Postfach 360 205, 1000 Berlin 36

★ ★ ★ **ESCHCOMP-SYSTEM** ★ ★ ★
EINNAHME/ÜBERSCHUSSRECHNUNG
sehr bedienungsfreundlich für Ärzte-Masseure u. alle freien Berufe.
20 Einnahmekont. 120 Ausgabekont. auf Wunsch bis insg. 500 Konten.
Ko. frei definierbar, b. 999 Bu/M.
Kostenstellen-Auswertung ....
Umsatzsteuervoranmeldung ....
Steuersätze frei def. und, und ......
1 / 2 Laufwerke einstellbar ....
für alle **SCHNEIDER Computer** ....
und für **VORTEX LW 5,25"** ....
DM 288,— incl. MwSt.
LEHRER – SCHULVERWALTUNG Progr.
für **SCHNEIDER Comp** mit Disc. 3"
Grundsch./Fachschu./Notenverw.
Zeugn. drucken mit Bemerk, (Bay)
Soziogramm, Quali und, und .....
Paukerpgr.,Wochenplan, Mathetr.
Info anfordern.
DM 398,— incl. MWSt.
VERSICHERUNGS-AGENTUR-Programm
bis 1000 Datensätze p/Disk, Alters-Berufsgruppen-, Terminaktionen, mit Terminkalender, Überweisungsträger drucken, Maske frei definierbar, SCHNEIDER 3" u. VORTEX 5,25"
DM 358,— incl. MwSt.
1 / 2 Laufwerke einstellbar
SUPER LOTTOPROGRAMM für
SCHNEIDER 3" u. VORTEX LW 5,25"
Info anfordern.
mit allen Schikanen **DM 99,—**
STAR Drucker NL 10 mit Modul nach Wahl **DM 898,—** incl. MWSt.
Wir haben über 500 Programme für SCHNEIDER Computer – Liste anfordern gegen Freiumschlag oder 2,– DM in Briefmarken bei
ESCHCOMP-SYSTEM, E. Scheithauer
8391 Perlesreut / Tel.: 08555/1390
Händleranfagen erwünscht
G

Viele preiswerte Spiele, Mathe, Anwend. (z.B. MINIVISICALC) von 2 – 28 DM!
Katalog geg. 1,– DM von
SCHNEIDERSOFT Andreas Wagner,
Gartenstr. 4, 8201 Neubeuern

ZYKLUS – bekannt aus Rundfunk und Presse – ist das ERSTE Prgrm. zur Familienplanung.
Jetzt für alle CPC-Rechner!
Disk 3" od. VDOS 2.0-Disk 5 ¼".
Ausf. Handbuch; kompl. DM 69,–.
★ INFO/BEST.: R.+K. Schäfer ★
★ Postf. 3224, D-5800 Hagen 1 ★

Original Textomat / Data Becker
Diskette + Handbuch, ungebraucht
VB: 75,00 – Tel.: 02241/83168

Konto 2000-Geldv. C 49,–/D 59,–
Star-Tex Textverarb. Disc. 79,–
Text-Super-5 Textv. C 29,–/D 39,–
Economic-Car C 29,–/D 39,–
Börsenspiegel C 29,–/D 39,–
Informationen: Th. Wieland
(C = Cass.) Falkertstr. 71
(D = Disc.) D-7000 Stuttgart-1

Verk. 12 tolle Spiele
zu Niedrigstpreisen (6 – 18 DM)
☎ 05661/3673

RIESEN-AUSWAHL
Spiel- und Anwendersoftware
Liste anfordern. M. Potthoff
4543 Lienen – Postfach 1290

BIORHYTHMUS
druckt Ihre natürlichen Lebensrhythmen als Sinuskurven oder Biozahl. m. Erklärg. f. 1 – 365 Tage.
Dieses Superprogramm schrieben Biorhythm. u. Grad. Ing. f. d. JOYCE.
Einführungspreis nur DM 74,85
Klaus R. Junker,
Postfach 1718,
6380 Bad Homburg v.d.H.

★ **NEU** ★ ★ **CHEMIE** ★ ★ **NEU** ★
Datenb. /m. Ch.-Test / Lernsoftw./ PSE/Atommodell/Elektronensyst.
4 Programme (!m. als 80 KB!) 30,– K / 40,– D / Typ? / Info 2,– Bfmk.
★ U. Hox, Ackerstr. 65, 4190 Kleve ★

Börsenprogramm für Profis
Chart, Zoom, gl. Durchschnitt, Linien, Balken, Überlagerung, Optionsscheine, Seriendruck. Info 1,– DM, Disk 200,– DM, Winfried Schmitt, Enzianweg 24, 8134 Pöcking

Fragen Sie uns nicht, wieso wir so billig verkaufen! Z.B.:
3-D Clock Chess f. JOYCE 62.90
Heroes of Karn f. JOYCE 59.90
Turbo-Adress f. JOYCE nur 127.90
Und noch mehr für die CPC's, z.B.:
Frank Bruno's Boxing 12.00
Dun Daragh 14.00, Slapshot 14.00!!
SUNSHINE-Software · A.d. Schilde 14 ·
5270 Gummersbach · Tel. 02261/75752
G

Gewerbliche Anzeigen sind mit „G" gekennzeichnet.

Landw. Spitzenprogramme
für CPC 6128
Schlagkartei, Sauenplaner usw.
Datenservice Wachendorf-Agrar
2814 Engeln 30
Info gegen 3,– DM Schutzgebühr G

Original Star Writer I
incl. Kopie DM 160,–; T. 07621/82364

30.000 DM für eine PC-Lösung?
300 DM mit Schneider CPC!
Qualitätssoftware für das Team vom Bau:
Massen- u. Flächenberechnung nach REB; Tiefbau; Statik u.a.
★ Info/Best.: Dipl.-Ing. Schäfer ★
★ Postf. 3224, D-5800 Hagen 1 ★

★ ★ ★ **CPC-Software** ★ ★ ★
Billige Software für Ihren CPC gibt es bei: TB-Software, Postfach 38, 7454 Bodelshausen

Biete an Hardware

Joyce-Speichererweiterung
512 KB nur DM 135,–, Lieferung gegen V-Scheck. Krawczyk-Versand, Pf. 280313, 1000 Berlin 28
G

CPC 464 Grün + Drucker NLQ 401 + Software + Data Becker-Lit. VB 1200,–; Tel.: 09252/6626

CPC 464 Farbe, DDI, Vortex SP512 Turbo Pascal, v. Bücher, Softw. NP: 4000,– f. VB 1790,– Tel.: 07182/2290

Verkaufe: CPC 464 + GT64 (Grün) + Drucker NLQ 401 DM 850,– Telefon: 0711/682053

CPC 464 Grün + DDI-1 + Disk + Bücher+Anw.-Soft.+CPC-Int.-Hefte ab 06/85. NP ca. 2100,– DM, 1a-Zust. kompl: VB 1400,– DM, Einzeln: VB Tel.: 02628/2420 ab 18 Uhr

CPC 464 Grün DM 560,–, M . Voorwold, Kruppstr. 98, 4000 Düsseldorf 1

Verk. NLQ 401 + Traktf. + Farbband, Roth. 2. LW 5.25" je DM 500,– Software VB, Tel.: 06103/25341

CPC 464 Color + Lightpen + 5 Bücher + Joy + Elite + Hexenküche + 4 Sonderhefte + Schneider Int. ab 7/85 – 7/86 und anderes Zubehör.
Neupreis – 2100,– DM
VHB kplt. – 1600,– DM
Telefon: 07225/1018 ab 18 Uhr

4-Farben-Drucker-Plotter ★ NEU★
Mit Centronics, auch als elekt. Reiseschreibmaschine DM 450,–
☎ 07681/1736

CPC 6128 Grünmonitor n. 3 Mon.
alt DM 1200,– ☎ 06192/5731

Drucker NLQ-401 + Ersatzband.
Ein Jahr alt. Für 490,– DM zu verk.
☎ 02136/31773, ab 19.00

CPC 464 Grün + DDI-1 VB 1250,– nur ausprobiert - T.: 0201/596692

Zweitfloppys f. alle CPC's anschlußfertig. 5" 390,– DM. 3" 260,– DM. versch. Hardware. Info W. Geiselhart, Fürst-Friedrich-Straße 5, 7482 Krauchenwies 1, T.: 07576/7170

Joyce Plus 3 Mon. m. Gar. und viel Software; Tel.: 06221/767470

Verkaufe CPC 464 DDI je 450,– DM
Wordstar 110,–, Devpack 80,– org. mit Handbücher. Für Spiele und Literatur. Info anfordern, bitte Freiumschlag. K. Tripke, Am Wedemarkbad 15, 3002 Wedemark 1, Tel.: 05130/1076

CPC 464-Verkauf Komplettanlage
CPC 464 mit Farbmonitor + Bücher, Diskettenlaufwerk DDI-1 mit original Texpack + Topcalc
Drucker: Star Gemini-10X + Kabel VB DM 2200,–, auch Einzelverkauf. Nach 17 Uhr - Chiffre: 090786

CPC 6128 (grün) + Wordst. +Lit. + Disk. für 1499,– ☎ 0941/27483

SP-64/256 Aufrüstsatz DM 64,–
2. Laufwerk f. **VORTEX F1** DM 359,–
Günstige Speicher-IC's und Floppys. Tel.: 089/806823, 17 – 19 Uhr
G

Floppy DDI-1 498,– DM, FD-1 398,– DM, Kontroller einzeln 229,– DM. Manfred Kobusch, Bergenkamp 8, 4750 Unna, Tel.: 02303/13345

★ ★ **CPC 6128 mit Grünmonitor** ★ ★
6 Mon. 1100,– DM. Datenrecorder, 14 Disk, Tasword, Wordstar: DM 250,– Alles zusammen 1350,– DM
☎ 089/7001776, Stefan Ries

Verkaufe CPC 664 (grün) mit Vortex Speichererweiterung SP 64 + Literatur + Software.
Preis: 700,– DM, ☎ 02274/6141

464 +Farb-Mon. 850,–, T. 0531/346248

Verk. Drucker 401, ☎ 04349/1222

CPC 464-, CPC 664-, CPC 6128-User
Lichtgriffel mit Programmen und dt. Anleitung nur DM 49,–
Versand gegen Scheck/Nachnahme
Info gratis! Fa. Schißlbauer
Postfach 1171S, 8458 Sulzbach
Tel.: 09661/6592 bis 21 Uhr G

CPC 464 Green Monitor, Vortex Doppel-Floppy, Joystick gegen Gebot zu verkaufen. Köln, Chiffre: 100786

Verk. wegen Systemwechsel
CPC 464 m. GT64+DDI-1+NLQ 401 +Lightpen+DEVPAC+Firmware+ Software+Bücher u. Zeitschr. VB 2700,–, NP 3700,–; Tel.: 0671/36563

Joyce-PCW 8256 mit Software
Multiplan - DR GRAPH - VB 1500,– DM; Telefon: 02361/12166

CPC 464, Farbe. ☎ 02154/7839

★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★
★ **HELISOFT** ★ **Tel.: 05609/9838** ★
★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★

★ CPC 6128 grün	975,– ★
★ CPC 6128 farbe	1645,– ★
★ CPC 464 grün	649,– ★
★ DDI 1	475,– ★
★ Zweitlaufw. 464	249,– ★
★ Joyce 8256	1599,– ★
★ 512 KB 8256	129,– ★
★ FD 2 8256	599,– ★
★ Joyce 8512	2129,– ★
★ DMP 2000	659,– ★

★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★
★ ADREVAMAT + die Adreßver- ★
★ waltung für den Joyce. ★
★ Superschnell, enorme ★
★ Kapazität 89,– ★
★ DRUCKERBOX zwei Drucker ★
★ an einem Rechner oder zwei ★
★ Rechner an einem Drucker 89,– ★
★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★
★ Alle anderen Artikel ähnlich ★
★ günstig. Versandzentrale ★
★ Tel.: 05609/9838 ★
★ Ladenverkauf, Friedrichstr. 18, ★
★ 3500 Kassel G ★
★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★ ★

■ ■ ■ ★ **SP 512 RAM-KIT** ★ ■ ■ ■
Aufrüstkit für Vortex SP64 auf 512 KByte RAM, nur 198,– DM
Eichbauer, T.: 09135/1252 ab 14 Uhr
G

CPC 464 Color+NLQ 401+Joy+Abdeckhaube+v. Literatur+Software ★ zu verk. für **VHB 1800,– ★ ☎ 06758/8281** ab 19.00 Uhr, Klein, 6551 Bockenau

CPC 464 Grün 3 Mon. nur 450,– DM
☎ 0651/10809

CPC-Bücher und Software zum halben NP
☎ 0651/10809

Power-CPC; IBM-Gehäuse; 2 Laufwerke; Color; 512KB; NLQ-Drucker; 26 Bücher; 130 Disketten + Box; sämtl. Zeitungen; viel Zubehör; sonst. NEU: ca. 12000,– DM; VB: 4500,– DM; Tel.: 0221/536739

Verk. CPC 464 + Floppy + umfangr. Lit. + Prog. für DM 1350,–
☎ 02382/3420

SUPERANGEBOTE!!!

3"-Disketten 10 St. nur	DM 59,–
Drucker CPA-80GS	DM 598,–
CPC 6128 Grün!!	DM 948,–

Computer*Versand*Trier* Pf. 2106, Tel.: 0651/16366 Info anfordern!
G

Höchstpreise für Ihr Altgerät
zahlt *Computer*Versand*Trier
Postfach 2106, Tel.: 0651/16366
G

CPC 464 Color neuwertig mit Zubehör für VB 900,– abzugeben.
☎ 0651/10809

Farb-CPC, DDI-1, NLQ 401, 2 Joysticks, 5 Bücher und Software DM 2000,–
☎ 0531/872222

JOYCE 5 ¼ Zoll Floppylaufwerk (EXT.)
zu verkaufen, 1 MB. Tel.: 06173/5034

JOYCE Plus 512K NEU + Software 2100,– Tel.: 0906/6784 nach 17 h

6128 mit Grünmonitor, Schneiderfloppy FD 1, Schneiderdrucker DMP 2000, 2 Joystick, Data-Recorder, CPC-Station, Netzteil MP-2, RS 232-Interface mit Netzteil, mit umfangreicher Software wie: Compack, Multiplan, WordStar, dBase II, Star Writer, Schneider-Basic-Kurs, Cumputer-Kurs, 3D-Schach, Spielesammlung mit 20 Spielen etc. mit ca. 25 Disketten, Markt & Technik Handbüchern, CPC-International komplett, sowie sonstige Literatur, Handbücher und Fachzeitschriften; alles mit Garantie, Jan '86 gekauft.
Neupreis 6.140,00 DM abzugeben für 3.800,00 DM.
☎ 09776/247

Suche Software

Suche dringend Schulsoftware
für CPC 6128
(Stundenplan - Schülerverwaltung - Notenverwaltung u.ä.)
Angebote an:
V. Strehlow,
Dammstr. 42, 2300 Kiel-1

Statistik (t-Test etc.) für Joyce
ges. Goetze H., Sedanstr. 19, 2000 HH 13

Fortran für CPC 664 Gesucht!
Chiffre 260686

Suche für Joyce PCW 8512
Terminprogramm
Tages-Wochen-Monats-Jahresterm.
G. Rockenfeller, Haarweg 63, 5870 Hemer, Tel.: 02372/1853

Suche Programme für Joyce
spez. f. Büro u. Kleingewerbe
Rechnungen, Buchhaltung etc.
Tel.: 04221/87175 ab 1.8., 19 Uhr

Suche für Joyce
alle Arten von Software.
Angebote mit Programmbeschr. an R. Wiedersich, Hainstädter Str. 42, 6452 Hainburg 2

Überweisungsdruck-Programm mit OCR-A1 Zeichensatz für 464 + DMP 2000 gesucht. Angebote: Thomas Winiarski, Pf. 305808, 2000 HH 36

Suche Software für CPC 6128
A. Hecht, Fachenfelder Winkel 4
2105 Seevetal 3

Russische Schriftzeichen
für Joyce gesucht: Drucker und Bildschirm
Telefon: 02101/43359

WordStar origin. kpl. für 6128
3"-Disk., Tel.: 0681/399821 Bonner

Suche Hardware

Suche DDI-1. ☎ 05346/1331.

Fortran für CPC 664 gesucht!
Chiffre 260686

Suche Drucker für CPC 464
G. Klemp, Apfelstieg 1, 2160 Stade

WANTED! Seikosha GP550A od. CPC.
W. Lack, Sieringstr. 29, 6230 Ffm

Verschiedenes

!!!!! BESUCHEN SIE UNS !!!!!
ELEKTRONIK & COMPUTERTAGE SAAR
Verkaufs-Informationsmesse 5.–7.9.
Kongresshalle Saarbrücken,
Info: PF 101260, 6620 Völkingen

Antireflex- und Kontr.-Scheibe für Schneider CPC 32,— DM
Videodatei-Programm a. Systeme CPC,IBM u. Kompat. (Disk) 48,— DM
IVS-D. Kamp, Köllmannstr. 13
4300 Essen 14, Tel.: 0201/594323

Etikette genau passend für 3"-Disketten
Bedruckt mit A- u. B-Seite.
20 Stck. nur 10,– DM inkl. Porto.
Nur gegen Vorkasse Schein/Scheck.
E-H-E, Schroerstraße 34,
390 Gladbeck

Tausch

Tausche super Spiele für Schneider CPC 464. Alles nur auf Kassetten.
Zuschriften an: Jürgen Mückstein,
Marienbader Str. 16,
8851 Mertingen.
Jede Kassette wird beantwortet.

Software-Tauschpartner gesucht
nur Cassetten für CPC 464 sowie 800XL und C64. Listen erbeten an Johann Feddermann, Breiter Gang 8, 2000 Hamburg 36

Colour-Monitor CTM644 gegen GT-65 + Ausgleich Tel.: 05323/40961

**Das ist Ihre Chance...
schon eine Kleinanzeige bringt oftmals großen Erfolg und hilft neue Kontakte knüpfen.
Nutzen Sie unser Angebot und profitieren Sie von der Tatsache, daß unsere Zeitschrift „Schneider CPC International" jeden Monat von mehreren zigtausend Computer-Interessierten gelesen wird.
Möchten Sie etwas verkaufen, tauschen oder suchen Sie das „Tüpfelchen auf dem I" – dann sollten Sie die eigens hierfür bestimmte Bestellkarte im Heft ausfüllen und an unseren Verlag absenden.
Ihre Annonce erscheint dann in der nächsterreichbaren Ausgabe.**

Wir möchten ausdrücklich darauf hinweisen, daß wir keine Anzeigen veröffentlichen, aus denen ersichtlich ist, daß es sich hierbei um Veräußerungen von Raubkopien oder dergleichen handelt.

Des weiteren machen wir darauf aufmerksam, daß indizierte Computerspiele nicht in Form von Anzeigen beworben werden dürfen.

Die Redaktion

C'86 im Zeichen des Handwerks

Computer-Ausstellungen sind in der Regel immer Anziehungspunkte der Massen. Die C'86 in Köln hatte vom 12. - 15. Juni ihre Pforten geöffnet und glänzte durch die Abwesenheit vieler namhafter Aussteller sowie durch ein nur schwer zu durchschauendes Konzept.

Schon am ersten Ausstellungstag gab es die ersten Mißverständnisse, was den Informationsfluß Veranstalter - Publikum anbetraf. Während vor den Toren der Kölner Messehallen zahlreiche interessierte Besucher auf Einlaß warteten, zeigten sich Aussteller und Veranstalter überrascht. Der Grund ist denkbar einfach; wurde doch der erste Ausstellungstag ausschließlich dem Fachhandel und der Presse gewidmet, nur gewußt hatte dies niemand!
Da war es auch nicht verwunderlich, wenn dem einen oder anderen Aussteller ein müdes Lächeln über die Vorfälle im Gesicht stand; aufgeregt hat sich wohl niemand und aufregend war es dann schon gar nicht.
Der Gang durch die beiden Ausstellungsetagen verlief zumindest für uns ohne große Zwischenstops, es war schlichtweg eine Enttäuschung.
Ganz eindeutig dominierten die Anbieter, die Hard- bzw. Software für das Handwerk zeigen konnten; vom Programm für Maler bis hin zu ganzen Branchenlösungen reichte die dort gezeigte Palette.
Dem interessierten Schneider-Anwender konnte da schnell die Lust vergehen. Nur wenige Aussteller waren mit Produkten für die CPC's bzw. Joyce bestückt.

Neues Diskettenformat

Am Stand von fischertechnik computing war zu erfahren, daß die Software für die Schneider-Baukästen in Kürze auch auf Diskette angeboten wird, so daß auch die 664- und 6128-Besitzer in den Genuß dieser Hardware-Erweiterungen kommen.
Der Sybex-Verlag präsentierte seine komplette Buch- und Softwarereihe, die inzwischen auf ein ansehnliches Potential für Schneider-Computer angewachsen ist. Hier wurde uns die Star-Datei angekündigt, die das Programm Star-Texter in Kürze um eine weitere Anwendung ergänzen wird. Des weiteren war beim Blick in das reichlich gefüllte Bücherregal festzustellen, das Sybex die Bücher zunehmend mit einem Hardcover versieht, was zweifelsohne dem Aussehen nach dem täglichen Gebrauch zu Gute kommt.
Überraschend verlief dann der Besuch am Stand von datatech. Hier wurden Disketten "made in Germany" angekündigt. Laut Auskunft ist demnächst mit der Auslieferung von 3"-Disketten zu rechnen, die zwischenzeitlich nur sehr schwer erhältlich waren.

Zusätzlich will diese Firma auch Disketten im 1 1/2"-Format herstellen, allerdings fehlen dazu bisher die entsprechenden Laufwerke. Diese kleinen Scheiben sollen eine Speicherkapazität von 1MB besitzen, was zweifelsohne eine revolutionäre Entwicklung einleiten würde. Abzuwarten bleibt, für welches Format sich die großen Computerhersteller entscheiden werden; zur Zeit scheint sich der Trend in Richtung 3,5" zu bewegen.
Höhepunkt der C'86 war, was jedenfalls das Zuschauerinteresse anbetraf, die parallel stattfindende Computerschach-Weltmeisterschaft, die bereits zum fünftenmal ausgetragen wurde. Hier mußte sich Vorjahressieger Mephisto mit zahlreichen Konkurrenten auseinandersetzen, das Ergebnis stand zu Redaktionsschluß noch nicht fest und wird später nachgereicht. Abschließend bleibt zu bemerken, daß viele führende Computerfirmen nicht anwesend waren, und damit konnte man den Eindruck nicht verwehren, daß Messetermin und -Konzept nicht glücklich gewählt waren.
Es kann nur noch besser werden, und damit dürfen wir auf die C'87 mit einiger Spannung warten.

(SR)

Aktuelle CPC Buchhits



Ein Muß für jeden, der sich professionell mit dem CPC 6128 oder dem CPC 664 beschäftigt. Einführung in das System, den Prozessor, das Gate Array, den Video-Controller, den Schnittstellenbaustein 8255, den Soundchip, die Schnittstellen. Mit Disassembler und ausführlichen Kommentaren zu den Routinen von Interpreter und Betriebssystem. Ein Superbuch, wie alle Titel der INTERN-Reihe!
CPC 6128/664 Intern
456 Seiten, DM 69,–



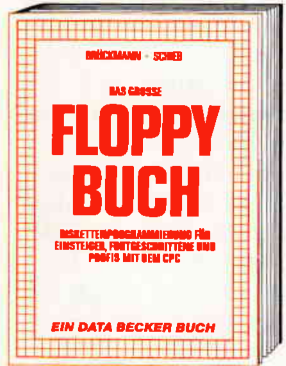
Wollen Sie in BASIC wie ein Profi programmieren? Dieses Buch macht es Ihnen leicht. Themenbereiche: Variablen, Zahlensysteme, Bits und Bytes, Tokens, Stringbearbeitung, Sortierung, Laufschrift, selbstdefinierte Zeichen, Windows, Rundungen, Fehlerbearbeitung, Kopierschutz, Grafiken, Joystick, Soundprogrammierung, relative Dateien u.v.m. Viele Beispielprogramme finden Sie in den entsprechenden Kapiteln.
Das große BASIC-Buch zum 6128
276 Seiten, DM 39,–



Der 2. Band CPC Tips & Tricks ist für alle CPC Besitzer interessant. Ob sie nun einen 464, 664 oder 6128 besitzen! Aus dem Inhalt: Menuegenerator, Maskengenerator, BASIC-Befehlserweiterungen, Programmierhilfen wie Dump, BASIC-Zeile von BASIC aus erzeugen, wichtige Systemroutinen und deren Nutzung, Beschleunigung von Programmen u.v.m. Wer noch mehr über seinen CPC wissen will, der kommt an diesem Buch nicht vorbei!
CPC Tips & Tricks Band II
250 Seiten, DM 39,–



Erlernen Sie die Maschinensprache Ihres CPC. Von den Grundlagen der Programmierung über die Arbeitsweise des Z80-Prozessors und einer genauen Beschreibung seiner Befehle bis zur Benutzung von Systemroutinen ist alles ausführlich und mit vielen Beispielen erklärt. Alle Hilfsroutinen, wie Assembler, Disassembler, Monitor und Einzelschrittsimulator, sind als Listings zum Abtippen enthalten. So wird der Einstieg leicht gemacht.
Das Maschinensprachebuch zum CPC
333 Seiten, DM 39,–



Einsteiger und Profis erfahren alles über die Arbeit mit der Floppy. Nützliche Routinen zur Verwaltung der Floppy, wie eine komfortable relative Dateiverwaltung, ein Disk-Monitor und ein Disk-Manager sind als Listings zum Abtippen enthalten. Eine Fundgrube verschiedener Programme und Hilfsroutinen mit ausführlicher Dokumentation der ROM-Einsprungadressen. Pflichtlektüre für jeden Anwender. Stark erweiterte und überarbeitete Neuauflage.
Das große Floppy-Buch zum CPC
422 Seiten, DM 49,–



Endlich CP/M beherrschen! Von grundsätzlichen Erklärungen zu Speicherung von Zahlen, Schreibschutz oder ASCII, Schnittstellen und Anwendung von CP/M-Hilfsprogrammen. Für Fortgeschrittene: Fremde Diskettenformate lesen, Erstellen von Submit-Dateien u.v.m. Dieses Buch berücksichtigt die Versionen CP/M 2.2 und 3.0 für Schneider 464, 664 und 6128.
Das CP/M-Trainingsbuch zum CPC
260 Seiten, DM 49,–



DFÜ für Jedermann mit dem CPC bietet eine ausführliche und verständliche Einführung in das Gebiet der Datenfernübertragung: was ist DFÜ, BTX, DATEX, Mailbox. Alles über Modems und Koppler. Begriffserklärung: Originate, Answer, Half-Duplex usw. Eine serielle Schnittstelle am CPC, RS 232/V.24 simuliert, Mailboxsoftware – selbstgestrickt, Postbestimmungen u.v.m. Steigen Sie mit diesem Buch in die Welt der Datennetze und Datenfernübertragung ein. **DFÜ für Jedermann zum CPC**
303 Seiten, DM 39,–



Speziell für den Hobbyelektroniker, der mehr aus seinem CPC machen möchte! Von nützlichen Tips zur Platinenherstellung über Adreßdecodierung, Adapterkarten und Interfaces bis zu EPROM-Programmierboard und -Programmiernetzteil oder Motorsteuerung für Gleich- und Schrittschaltmotoren werden machbare Erweiterungen ausführlich und praxisnah beschrieben. Am besten gleich anfangen!
CPC Hardware-Erweiterungen
445 Seiten, DM 49,–



LOGO, eine Sprache, die immer beliebter wird für alle CPC- und JOYCE-Anwender. LOGO ist einfach zu erlernen, aber vielseitig in der Programmierung. Das Buch befaßt sich unter anderem mit folgenden Themen: Rechnen mit LOGO, Grafikprogrammierung, Wörter- und Listenverarbeitung, Prozeduren und Rekursionen, Sortierroutinen, Maskengenerator, Datenstrukturen und Künstliche Intelligenz.
Das große LOGO-Buch zu CPC und JOYCE
ca. 300 Seiten, DM 39,–



Dieses Buch führt Sie Schritt für Schritt in die Benutzung des Joyce ein. Diese Einführung geht von der Installation der Geräte über eine Einleitung in LocoScript bis hin zur Programmerstellung in BASIC und LOGO. Auch die wichtigsten Befehle des Betriebssystems CP/M 3.0 werden leicht verständlich beschrieben. Der ideale Einstieg mit dem Joyce!
Joyce für Einsteiger
248 Seiten, DM 29,–



Das Superbuch für jeden Joyce-Anwender. Alles über die Textverarbeitung LocoScript und über das Betriebssystem CP/M; Bedienung, Anwendung und Lösungen für dBase, Multiplan und WordStar; BASIC-Routinen wie Menü- und Maskengenerator und rekursive Grafikprogrammierung in LOGO.
Das große JOYCE-Buch
Hardcover, 424 Seiten, DM 59,–

Und wo informieren sich CPC-Anwender über News & Trends, neue Software, neue Computer und aktuelle Tips & Tricks? In der

DATA WELT 7 · 8/86

Wo denn sonst!
DATA WELT 7 · 8/86
jetzt am Kiosk.

BESTELL-COUPON

Einsenden an: DATA BECKER · Merowingerstr. 30 · 4000 Düsseldorf 1

Bitte senden Sie mir:

zzgl. DM 5,– Versandkosten

☐ per Nachnahme ☐ Verrechnungsscheck liegt bei

Name

Straße

Ort

DATA BECKER

Merowingerstr. 30 · 4000 Düsseldorf · Tel. (0211) 310010

Gamers Message

Obgleich viele unserer Leser derzeit lieber irgendwo in der Sonne liegen und sich rundum knusprig braun braten lassen, meldet sich hier die Gamers Message mit Tips und Tricks für alle diejenigen, die ein heißes Computerspielmatch einer Bräunungs-Saison vorziehen.

ANDROIDENSCHAFE UND DEREN BÄNDIGUNG
Den Anfang machen hier die zwanzig Codewörter für die einzelnen Level bei "One Man and his Droid".

1. Kein Name
2. EMPIRE
3. PREDATORY
4. RUMINATE
5. RYEGRASS
6. VACUUM
7. VAMPIRE
8. RAGOUT
9. GRAIN
10. AASVOGEL
11. BLIZZARD
12. CLOCHE
13. COLANTER
14. ECTOPLASM
15. ECOLOGY
16. FEROCIOUS
17. FETLOCK
18. GOOSBERRY
19. GRAVITATE
20. UPANDAWAY

Mit diesen Paßwörtern kann nun jedermann in jedem Level dieses Spiels beginnen, ohne sich vorher durch die davorliegenden Stufen zu kämpfen.

UNSTERBLICHE SUCHER
Wenn Sie sich die Mühe machen und anstelle des eigentlichen Loaders bei "Finders Keepers" das nachfolgende kleine Programm zu installieren, haben Sie keinen Ärger mehr mit zu wenig Energie. Die steht Ihnen dann nämlich unbegrenzt zur Verfügung.

```
10 openout "d
20 memory &7ff
30 load"",&800
40 poke&20ce,0
50 call&800
```

Außerdem noch eine kleine Auflistung der Gegenstände, die mit einem anderen in Verbindung gebracht, einen neuen ergeben. "Pile of Mud" + "Spark of Life" = "Mud Monster", "A Blacksmith" + "A broken Sword" = "Excalibur", "A empty Bottle" + "A Model Boat" = "A Bottle Ship", "A Lead Bar" + "Sages Stone" = "A Gold Bar". Zu guter Letzt können Sie sich aus "Salpetre + "Sulphur" + "Charcol" + "Magic Flame" Schießpulver mixen, und damit die Katze vor dem Ausgang wegsprengen. Mit dem Miniprog und den Tips sollte es eigentlich jetzt jedermann schaffen, dieses Spiel bis zum Ende zu bringen.

PROFESSOREN, GOLDENE ÄPFEL UND ANDERES ZEUG
Für ein recht neues Spiel, nämlich "Contraption" von Audiogenic, haben wir hier ein vergleichbares kleines Programm, das wie das von "Finders Keepers" gehandhabt wird.

```
10 openout"d"
20 memory 1999
30 load"",15000
40 call286500
50 load"",15700
60 poke&5ec9,0
70 poke&5eca,0
80 poke&5ecb,0
90 poke&5ecc,0
100 call23700
```

COMETENFINISH
Als kleine Lockerungsübung hier ganz am Rande ein Minitip zu "Comet".
Um ohne große Schlachten in das letzte Bild zu gelangen, brauchen Sie lediglich das Spiel zu "pausen", und dann gleichzeitig die Tasten 0 und 9 niederzudrücken.

DAS PLUS VON SORCERY+
Eines der besten CPC-Spiele ist nach wie vor Sorcery und das "follow up" Sorcery+. Daß es in diesem Action-Adventure so manches Rätsel zu lösen gilt, versteht sich von selbst. Stefan Achilles ist es gelungen, den üblen NECROMANCER zu besiegen. Im folgenden verrät er nun das "know how".

Zuerst sollte man, wenn Sorcery+ gelöst ist, einen "GOBLET OF WINE" mitnehmen. Es erleichtert Sorcery+ unheimlich. Jetzt betritt man die linke obere Tür und beginnt mit dem zweiten Teil des Spieles. Bei "NEAR THE GATEHOUSE" kommt man mit dem Goldenen Schlüssel, der vor der Tür liegt, herein. Dann ist man "IN THE GATEHOUSE", geht mit der "STRANGELOOP CASSETTE" in die linke untere Tür, jetzt befindet man sich "UNDER THE GATEHOUSE", befreit das Herz mit der "STRANGELOOP CASSETTE", bringt dieses in Sicherheit und holt den Silbernen Schlüssel. Dann geht man wieder heraus, in die obere rechte Tür und kommt dann "UNDER THE BELFRY" (mit dem Kreuz kann man einen von den beiden Dämonen töten). Jetzt geht man in die linke obere Tür, holt sich eine Glocke; dann muß man den ganzen Weg zurückgehen bis vor das Haus "NEAR THE GATEHOUSE". Man geht durch die untere linke Tür (Vorsicht vor dem Schlamm) und holt sich mit der Glocke die "COPY OF AMSTRAD USER". Dann geht man zurück und holt sich die "GLASS BOTTLE" und kehrt zurück, öffnet die Schlammwand. Danach wird der Silberne Schlüssel wieder geholt, mit ihm geht man durch die hinter der Schlammwand befindliche Tür. Dort kann man erst einmal wieder neue Energie tanken (zieht niemals ab). Jetzt holt man sich einen Stock "WOODEN CLUB" und geht durch eine der drei unteren linken Türen (mehrmals versuchen, Tür klemmt). Nun ist man "ON THE BRIDGE". Mit dem Stock können Sie nun die Schlammwand öffnen. Danach holt man sich einen weiteren Stock und geht in die linke Tür (Vorsicht, der Steg ist auf der anderen Seite gebrochen). Mit dem Stock berührt man die Pflanze und schon öffnet sich die Tür. In dem nächsten Bild sollte man nicht auftanken (zieht immer ab). Jetzt ist es an der Zeit, das Herz und die Copy of Amstrad-User nachzuholen. In diesem Bild "NEAR THE HIDEOUT" können Sie beide Teile liegen lassen und fünfmal durch die linke obere Tür gehen. Dabei besorgen Sie sich einen Silbernen Schlüssel. Dann müßten Sie bei "NEAR THE HIDEOUT" angekommen sein und zweimal durch die linke untere Tür gehen. Jetzt sind Sie "AT THE HIDEOUT ENTRANCE", gehen durch die rechte Tür "NEAR THE NECROMANCER" und nehmen die zwei Herzen.
Dann gehen Sie wieder nach rechts "THE MAIN HALL" und drücken den Feuerknopf. Jetzt holen Sie das dritte Herz und den "COPY OF AMSTRAD-USER" hierhin. Mit dem "COPY OF AMSTRAD-USER" geht man "AT THE HIDEOUT ENTRANCE" durch die Schlammwand (läßt sich nur mit GLASS BOTTLE öffnen). Dann sind Sie im "DESERTED CELLAR" und gehen durch die linke Tür (können vorher auftanken). Nun befindet man sich "JUST IN TIME" und holt sich "Roland". Dann verläßt man den Raum, geht in "DESERTED CELLAR" in die rechte untere Tür, wo man Roland per Feuerknopf ins Wasser wirft und ein viertes und letztes Herz erhält. Damit geht man wieder zur "MAIN HALL" und Sorcery+ wäre damit gelöst.

EUER HEINRICH

Joyce-Trainingsseminare

Seit der Markteinführung des Schneider Joyce sind nun einige Monate vergangen, etwa 30.000 Systeme sind inzwischen hierzulande verkauft worden. Ursprünglich als kompaktes Textsystem für Büroetagen angepriesen, stellte sich aber schon bald heraus, daß der Joyce weit mehr als nur eine komfortable Speicherschreibmaschine mit Monitor ist.

Pfiffige Anwender merkten schnell, daß mit dem Joyce eine preiswerte CP/M-Maschine gemeint war, die von Beginn an auf eine recht große Software-Palette zugreifen konnte.
Schwieriger war es dann doch, einer Sekretärin die, im Vergleich zu Ihrer gewohnten Schreibmaschine, recht seltsame Tastatur schmackhaft zu machen. Da eine Sekretärin bekanntlich das 10-Finger-Schreibsystem beherrscht und auch beim Arbeiten mit der Schreibmaschine keine zusätzlichen Funktionstasten à la Joyce benötigt, ist eine Umstellung unumgänglich.

Um ein solch revolutionäres Schreibsystem wie den Joyce an den Mann bzw. Frau zu bringen, benötigt man neben einem guten Händlernetz auch das entsprechende Fachpersonal in den einzelnen Verkaufsstellen.
Die Fa. Schneider beauftragte dazu im vergangenen halben Jahr das Institut Leidinger & Haupenthal mit entsprechenden Produktschulungen für Händler.
Diese Schulungen wurden in Tagesseminarform in Hotels der Novotelklasse mit großem Erfolg durchgeführt. Aufgrund einer starken Nachfrage, seitens der privaten Anwender,



hat das Institut entsprechende Seminare für Privatleute im Zeitraum Juli bis August geplant.
Diese Veranstaltungen finden bundesweit in folgenden Städten statt: Berlin, Hamburg, München, Frankfurt, Bremen, Hannover, Köln, Siegen, Dortmund, Stuttgart, Freiburg, Nürnberg, Mannheim, Saarbrücken und Trier.
In erster Linie geht es bei diesen Seminaren um das Textverarbeitungssystem Locoscript.
Der Teilnehmer soll nach Beendigung des Seminars den Joyce sinnvoll als Textsystem einsetzen können. Alle möglichen und erforderlichen Handgriffe, angefangen von der Formatierung von Disketten über die Besonderheiten von Locoscript zur Organisation von Texten, werden besprochen und systematisch trainiert. Des weiteren stehen Einsatzmöglichkeiten des Joyce im privaten, beruflichen und betrieblichen Bereich sowie ein grober Softwareüberblick auf dem Programm. Im Unterricht erhält jeder Teilnehmer ein 25 DIN-A4-Seiten umfassendes Nachschlagewerk, das u.a. einen Softwareüberblick beinhaltet und jeder Teilnehmer mit nach Hause bekommt.
Jedem Seminar-Teilnehmer steht ein eigener Joyce zur Verfügung, die Teilnehmerzahl ist auf maximal 12 Personen begrenzt.
Der Preis für ein Tagesseminar, das in der Regel am Sonnabend stattfindet, beläuft sich auf etwa DM 199,–.

Freizeit und Computer

Als Alternative zu den Tagesveranstaltungen werden neuerdings auch kombinierte Wochenendseminare angeboten, bei denen der Partner ohne Seminarteilnahme mitfahren kann. Hier steht die Freizeit eindeutig im Vordergrund. Die Unterrichtsdauer beträgt in diesem Falle 14 Stunden, ein Wochenendseminar dauert von Freitag nachmittag bis zum Sonntag und findet ohne Ausnahme im Schwarzwälder Hochwald statt.
Hier stehen mehrere Alternativen und Preisklassen zur Auswahl, der Preis richtet sich nach der Hotelwahl (DM 310,– bis DM 599,–).
Wie vom Veranstalter zu erfahren war, besaßen viele Teilnehmer keine oder kaum Kenntnisse über den Computer selbst. Daher wurde eine zusätzliche Veranstaltung ins Leben gerufen, die sich der Materie Computer annimmt und deren Arbeitsweise anschaulich und verständlich darstellt. Diese Veranstaltungen sind entweder als Abendschul- oder Wochenendunterricht ausgelegt, nähere Informationen hierzu erhalten Sie direkt beim Veranstalter.
Im Gespräch mit mehreren Seminarteilnehmern konnten wir erfahren, daß man durchweg mit den Leistungen des Veranstalters zufrieden war und auch die vielfältigen Angebote großen Anklang fanden. (SR)

Inserentenverzeichnis Nr. 8



9/86

„Schneider CPC 9/86" erhalten Sie ab 27. August bei Ihrem Zeitschriftenhändler



Joyce-Software ist im Kommen! Wir haben alle Joyce-Programme für Sie zusammengetragen und stellen Ihnen diese in einer großen Marktübersicht vor.



Die Simulation ökologischer Systeme auf dem Computer ist eine Herausforderung für jeden Programmierer. Anhand des bekannten Wolf-Schaf-Gras-Modells läßt sich die Problematik anschaulich verfolgen.



Die serielle Schnittstelle wurde einem ausführlichen Praxistest unterzogen. Das original Amstrad-Interface überrascht durch einige Features, die das Arbeiten eines Programmierers wesentlich erleichtern.

Serie:

Der nächste Teil unseres beliebten Software-Experiments beschäftigt sich mit der Simulation ökologischer Systeme. Als Grundlage dient das Wolf-Schaf-Gras-Modell, das in eindrucksvoller Art und Weise die Einflüsse und Auswirkungen verschiedener Begebenheiten auf ein ökologisches System anschaulich macht. Schon jetzt möchten wir allen interessierten Software-Experimentlern diese Folge ans Herz legen, denn auch Sie können einiges zum Gelingen dieser komplexen Serie beitragen.

Joyce:

Joyce-Software auf einen Blick!
Unsere große Software-Übersicht zeigt Ihnen alle Programme, die zur Zeit für den Joyce erhältlich sind. Des weiteren stellen wir Ihnen das McLin-Programmgenerator-System vor, mit dem Sie auf einfache Art eigene Programme erstellen können.

Programme:

U.a. präsentieren wir Diskmat – das komfortable Diskettenarchiv!
Auf dieses Programm haben alle Diskettenbenutzer gewartet. Diskmat archiviert Ihre Programme durch einfaches Einlegen der Diskette – schon haben Sie Ordnung in Ihrer Programmsammlung.

Hardware:

SCHNEIDERWARE # 4
Kraftwerk für die Erweiterungen
- das universelle Netzteil im Selbstbau!

Die Schwächen der Stromversorgung des CPC, gerade in Hinsicht auf weitere Karten, werden hierdurch ein für allemal eliminiert.

Tips & Tricks:

Der Tip des Monats: Bildschirmkompressor
- sparen Sie Speicherplatz auf Ihrem Datenträger beim Abspeichern von HiRes-Bildern mit dieser tollen RSX-Erweiterung!

UDG-Designer
- der komfortable Zeichengenerator für alle CPC!

Super-Hilfsprogramm zum Definieren eigener Grafikzeichen bzw. kompletter Zeichensätze. Die entworfenen Zeichen werden gleichzeitig in allen drei Modi dargestellt, die entsprechenden Werte lassen sich in HEX,DEZ und BIN abrufen.
Der UDG-Designer ist komplett menuegesteuert und bietet zahlreiche Optionen.

CP/M:

Rando und die Gappys!
Wir vermitteln weiteres Wissen über die Handhabung von CP/M und vertiefen die Praxis im Umgang mit den Tools. Rando und die Gappys soll Ihnen demonstrieren, wie schnell Bewegungsabläufe unter CP/M programmiert werden können, selbst wenn man beim Programmieren nicht gegen die Konventionen der BDOS-Calls verstößt.

Test:

Auf dem CPC-Prüfstand mußten zwei Hardware-Produkte ihre Leistungsfähigkeit unter Beweis stellen. Getestet haben wir die original RS-232-Schnittstelle von Amstrad, die sowohl unter Basic als auch unter CP/M zu verwenden ist.
Auf Herz und Nieren geprüft wurde auch die Speichererweiterung von dk'tronics, die seit kurzem auch bei uns erhältlich ist.
Alles Wissenswerte dazu erfahren Sie in der nächsten Ausgabe.

Software Reviews:

Spiele:
Equinox
Biggles
Gunfright
Shogun

Anwendungen:
Turbo-Basic – der neue Compiler
A.D.A.M
Profi-Basic

Software von Lesern – für Leser

DIE GOLDENE

NEU JETZT NR.



Schatz
Als Hubschrauberpilot für besonders brenzlige Einsätze bekommen Sie den Auftrag, durch ein verzwicktes Höhlensystem bis zu einem legendären Schatz vorzudringen und mit diesem wieder heil zurückzukehren.
Aber Vorsicht ist geboten, der Schatz ist schwer und die Höhlen verwinkelt und eng.

Garten Manager
Als Gärtner hat man es nicht leicht. Versorgen Sie das Grünzeug in Ihrem Garten richtig, dann kann gar nichts passieren. Abgesehen davon, daß Sie überall das Unkraut jäten und die Blumen gießen müssen, geht fast alles wie von selbst.
Doch das Unkraut wuchert wie wild und die Blumen brauchen immer mehr Wasser. Haben Sie einen grünen Daumen?

7 Programme für CPC 464/664/6128

Die Bezeichnung »Goldene Sieben« steht künftig für gute Software, die von den Lesern unserer Zeitschrift erstellt worden ist, und nicht als Listing im Heft abgedruckt werden kann, da hierfür nur einbegrenzter Raum vorhanden ist, und Ihre "Schneider CPC International" kein reines Listingblättchen sein soll!
Jeweils sieben tolle Programme sind auf der Kassette bzw. Diskette für wenig Geld enthalten.

Preise:

Kassettenversion 20,– DM
3"-Diskettenversion 30,– DM
zuzüglich 3,– DM Porto/Verpackung
(Im Ausland 5,– DM)
Bei Nachnahme kommt noch die die Nachnahmegebühr hinzu.
Ihre Bestellung richten Sie bitte an:

DMV Verlag
– Leserservice –
Postfach 250
3440 Eschwege

Bitte Bestellkarte benutzen!

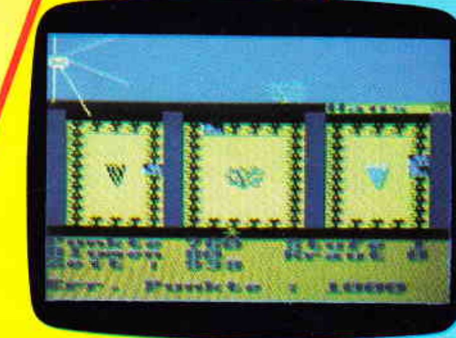
Berg der Monster
Grauenerregende Ungeheuer terrorisieren einen abgelegenen Landstrich in den Bergen. Als Lehensherr dieser Provinz bekommen Sie vom König den Auftrag, wieder Ruhe und Ordnung zu schaffen. Als Sie sich zu Ihrer Mission aufmachen, werden Sie von einem Schneesturm überrascht. Nach dem Unwetter stehen Sie ohne Ausrüstung und Hilfe alleine inmitten der Berge.
Gelingt es Ihnen trotzdem, Ihren Auftrag zu erfüllen?

Poker
Kartenspieler hergehört: wollen Sie ganze Nächte durchzocken, ohne Muttis Haushaltsgeld zu verspielen? Bitte sehr, unser Computerpoker macht's möglich. Gewinnen können Sie zwar nichts, aber trainieren läßt sich mit diesem Spiel umso besser.

Super Chance
Verwandeln Sie Ihren CPC in einen einarmigen Banditen. Mit dieser Spielhallensimulation holen Sie sich ein Stück Las Vegas in Ihr Wohnzimmer. Mit Jackpot und Superchance.

Captain Starships Test
Begleiten Sie Captain Starlight auf seiner Weltraumexpedition. Doch zuerst müssen Sie beweisen, daß Sie die notwendigen Qualitäten mitbringen. Ein kniffeliger Intelligenz- und Reaktionstest steht Ihnen bevor. Sind Sie in der Lage zu bestehen?

Oil Willi
Sie sind der nette Heizölhändler, der alle Bewohner seines Stadtviertels mit Heizöl versorgt. Aber die meisten Bestellungen sind sehr eilig. Und wenn Sie nicht schnell genug liefern, so bestellen sich die Leute ihr Öl eben anderswo. Da kann man schon ganz schön ins Schwitzen geraten.







7 Programme für CPC 464/664/6128

Auch weiterhin ist die "Goldene 7" Nr. 1 mit den Programmen Secret of Wizard, Turlen, Zodora, 3-D Labyrinth, Die alte Burg, Space-Race und Galaxis erhältlich.

Bitte benutzen Sie die Bestellkarte!